UNIVERSITY OF CALIFORNIA
RIVERSIDE

# Video Bioinformatics: Human Embryonic Stem Cell Analysis With Machine Learning

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering

by

Benjamin Xueqi Guan

December 2019

Dissertation Committee:
- Dr. Bir Bhanu, Chairperson
- Dr. Prue Talbot
- Dr. Hyle Park
- Dr. Yingbo Hua

The Dissertation of Benjamin Xueqi Guan is approved:

_____

_____

_____

Committee Chairperson

University of California, Riverside

# ACKNOWLEDGEMENTS

I thank my dissertation chair, Dr. Bir Bhanu, who never gives up on motivating me to finish my degree. I would like to thank Dr. Prue Talbot, Dr. Hyle Park and Dr. Yingbo Hua for being my committee members on my dissertation defense. I would also want to thank Dr. Gerardo Beni, Dr. Amit K. Roy-Chowdhury and Dr. Anastasios Mourikis for conferring my doctoral candidacy. I am grateful for the friends/colleagues who were part of my graduate life: Asong Tambo, Albert Cruz, Anthony C. Bianchi, Angello Pozo, Benjamin Sander, Giovanni Denina, Hoang T. Nguyen, John Gonzales, Jun wang, Nikki Weng, Yanping Chen and Zhixing Jin. I also want to thank Dr. Ninad S. Thakoor for the inspirational discussions as well as his support on my first paper. I am also extremely grateful for Nikki Weng and Dr. Sabrina Lin's willingness to collect data and make the CL-Quant software recipes for the projects. My research was supported by NSF-IGERT: Video Bioinformatics Grant DGE 0903667 and by Tobacco-Related Disease Research Program (TRDRP): Grant 20XT-0118 and by a TRDRP Postdoctoral Fellowship (20FT-0084). This dissertation was compiled from "Bio-Driven Cell Region Detection in Human Embryonic Stem Cell Assay", "Bio-Optimized Segmentation of Blebs in Human Embryonic Stem Cell Videos", "hESC Classification: Random Network with Autoencoded Feature Extractor" by B.X. Guan et al. © 2014, 2015 the IEEE.

To my parents and brother, I thank you for your supports and understanding on my pursuit for higher education. I also thank my wife and my son, Branderson, for your encouragements.

ABSTRACT OF THE DISSERTATION

Video Bioinformatics: Human Embryonic Stem Cell Analysis With Machine Learning

by

Benjamin Xueqi Guan

Doctor of Philosophy, Graduate Program in Electrical Engineering
University of California, Riverside, December 2019
Dr. Bir Bhanu, Chairperson

Human Embryonic Stem Cell (hESC) have a great potential for regenerative medicine to provide treatments for Parkinson's disease, Huntington's disease, Type 1 diabetes mellitus, etc. Consequently, hESC are often used as a model in the biological assay to study the effects of chemical agents on the human body. Video analysis plays an important role for biological assays in the field of prenatal toxicology and stem cell differentiation. This thesis introduces machine learning techniques for detection, segmentation and classification for hESC analysis. For the detection, a bio-driven algorithm was used to detect cell regions in hESC images. Cell region detection is essential in stem cell focused analysis. It can prevent background information from contaminating the analysis and put more emphasis on processing the cell region. For the segmentation part, a bio-inspired method was proposed for bleb extraction and analysis over time. Bleb formation is a strong health indicator of the stem cell undergoing chemical reactions. Therefore, it is significant to biologist to analyze the formation process over time. For the classification, a deep learning structure was built with both labeled and unlabeled hESC data to classify the six common classes in stem cell images. The six classes are: 1). cell clusters, 2). debris, 3). unattached cells, 4). attached cells, 5).

dynamically blebbing cells, and 6). apoptotically blebbing cells. Various results are provided on real video datasets collected using a phase contrast microscope and a Nikon Bio-station.

# TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

# CHAPTER 1 Introduction

Video Bioinformatics is an emerging field in providing solutions to biologists' need for faster and easier ways to analyze large volumes of video data. The biologists who study human embryonic stem cells (hESC) have to deal with stem cell videos every day, and the analysis of videos is a laborious manual process. It is important to use the information from time lapse videos to study hESC's behavior during exposure to various chemical agents. Most stem cell videos are taken with the phase contrast microscopy. It is challenging to analyze these videos automatically. The low signal to noise ratio (SNR) of the phase contrast images makes it hard to analyze the contents in the image. Consequentially, machine learning techniques such as segmentation, detection and classification are explored for automated and fast analysis in hESC. In order to have a sound system, modularization structure must be implemented. Modularized components allow for flexibility and adaptability in the system [1] [2] [3]. We consider detection, segmentation and classification as three separate modularized components or subsystems. Detection, segmentation and classification subsystems are addressed as individual chapters in this paper.

For detection component, we introduced a bio-driven algorithm that detects cell regions automatically in hESC images obtained using a phase contrast microscope. The algorithm uses both statistical intensity distributions of foreground/hESC and background/substrate as well as cell property for cell region detection. The intensity distributions of foreground/hESC and background/substrate are modeled as a mixture of two Gaussians. The cell property is translated into local spatial information. The solution

of the algorithm is optimized by parameters of the modeled distributions and cell regions evolve with the local cell property. This chapter validates the method with various videos acquired using different microscope objectives. In comparison with the state-of-the-art methods, the proposed method is able to detect the entire cell region instead of fragmented cell regions. It also yields high marks on measures such as Jaccard similarity, Dice coefficient, sensitivity and specificity. Automated detection by the proposed method has the potential to enable fast quantifiable analysis of hESC using large datasets which are needed to understand dynamic cell behaviors.

For segmentation component, we proposed a bio-optimized method for segmentation of hESC blebs in videos. Full bleb formation consists of bleb expansion and retraction. The bleb changes its size and image properties dynamically in both processes. The proposed method uses adaptive parameters in segmentation algorithms for bleb region detection during both processes. The adaptive parameters are found by using an optimization metric. The metric parameters are derived from model fitting of bleb area change over time for both bleb processes. Through model fitting, we observed that both processes showed different rates in change of area over time. Therefore, two models are used in the optimization metric. In comparison to complete manual analysis, the proposed method provides a fast and accurate method to analyze videos of bleb expansion and retraction.

For classification component, we present random network (RandNet) with autoencoded feature extractor for the classification of human embryonic stem cells. Automated understanding of hESC videos is essential for the quantified classification and

analysis of various states of hESC in experimental work. A video of recently plated hESC can be regarded as a set of frames that are composed of a total of six classes: 1) cell clusters, 2) debris, 3) unattached cells, 4) attached cells, 5) dynamically blebbing cells, and 6) apoptotically blebbing cells. This method focuses on using ensemble concepts and deep learning techniques to develop a classification model for hESC classification. The proposed method also compared with current state-of-art deep learning methods.

# CHAPTER 2 Related Works and Contributions

## 2.1 Cell Region Detections

In this section, we discuss detection algorithms. K-means algorithm and mixture of Gaussians using an Expectation-Maximization (EM) algorithm are widely used techniques for image segmentation. K-means segmentation algorithm by Tatiraju et al. [4] considered each pixel intensity value as an individual observation. It partitions these observations into K clusters in which each observation belongs to the cluster with the nearest mean intensity value [5] [6]. However, the method does not consider the intensity distribution of its clusters. In contrast, the mixture of Gaussians segmentation method using the EM (MGEM) algorithm proposed by Farnoosh et al. [7] depends heavily on intensity distribution models to group the image data. The MGEM method assumes the image's intensity distribution can be represented by multiple Gaussians [4] [8] [9]. However, it does not take into account the neighborhood information. As a result, segmented regions obtained by this above algorithms lack connectivity with the pixels within their neighborhoods. This lack of connectivity of a pixel with its neighborhood pixels is due to the following two characteristics of hESC images: i) an incomplete halo surrounds the cell body; ii) cell body intensity values are similar to the substrate intensity values [10].

## 2.2 Computational Models for Blebbing in Biology

In this section, we discuss the formulation of computational models for blebbing in biology. This idea on computation models gave rise to Chapter 4 on bio-optimized

segmentation of blebs in human embryonic stem cell videos. Charras et al. [11] [12] reasoned that blebbing depends on parameters such as pressure, membrane-cortex, adhesion energy and membrane tension of a cell. The plasma membrane of a hESC is attached under tension to a cortex of filaments. If the connection with the filaments is weakened, a bleb is produced by an event of pouring cytoplasmic fluid into the weakened region. When the growth of the bleb stops, the bleb either retracts or stays the same. If an actin cortex reforms under the bleb membrane, retraction is likely to occur and is driven by myosin-II.

StryChalski et al. [13] assumed that blebbing occurs due to detachment of the cytoskeleton from the plasma membrane, which produces a pressure-driven flow of cytosol toward the area of detachment and into the area of expansion. They proposed a computational model of blebbing based on the mechanics of intracellular fluid, the actin cortex, and the cell membrane. The model considers the bleb formation time as a function of parameters derived from cytoplasmic properties [13]. A similar model also has been proposed in [14]. Derived from the aforementioned concepts, we proposed a bio-optimized segmentation method for both cell expansion and retraction in video.

### 2.3 Human Embryonic Stem Cell Classification

In this section, we will solely focus on the classification component. There has been very limited work for building an automated classification system for stem cells in video with both labeled and unlabeled datasets [15]. Niioka et al. used convolutional neural network (CNN) to study cellular differentiation from myoblasts to myotubes [16]. Their classification model was built upon the concept that cellular morphology changes

during differentiation, and this feature was easily captured in stained fluorescent images. In addition, Xie et al. worked on fluorescent images with CNN for cell counting [17]. Although they have a successful experiment, but their classification problem was simple since their images contained only circular dots. Chang et al. also used CNN for human induced pluripotent stem cell regions classifications [18]. Their study focused on classifying cell cluster patterns. The dataset used in the works by Niioka [16], Xie [17] and Chang [18] came from experiments that use staining techniques; staining is a very intrusive technique to be used on cells for contrast enhancement. However, our hESC experiments were done without staining.

Similar work on stem cell classification with phase contrast images was proposed by Theagarajan et al. [19] [20]. They suggested using a generative method to train the network and classify real data. However, they did not consider realistic unlabeled data which can be efficiently generated for training; typical generative methods have huge computational cost for synthetic dataset generation as well as training with a large set of synthetic data. Therefore, this Chapter proposed on using the unlabeled data (without the use of generative methods) for model training and fine-tuning the model with labeled data.

We have made three contributions in this Chapter. First, we introduce the concept of creating a modularized system to automatically segment and classify hESC in video. Second, we introduce the concept of building feature extractor with unlabeled data. Third, we incorporate ensemble methodology into random network. Finally, we provide experimental results and compare them with state-of-the-art techniques.

# CHAPTER 3 Bio-Driven Cell Region Detection in Human Embryonic Stem Cell Assay

Human embryonic stem cells (hESC) are pluripotent cells derived from the inner cell mass of blastocysts, and in culture, they closely resemble epiblast cells of gastrulating embryos [21] [22]. Due to fact that hESC have the ability to self-renew indefinitely and to differentiate into all three germ layers (ectoderm, endoderm, and mesoderm), they are widely used in research designed to tap their potential for treating degenerative diseases. In addition, hESC provide one of the best models currently available for assessing the toxicity of environmental chemicals on prenatal development [23] [24].

Application of video bioinformatics tools to hESC problems can greatly accelerate research in both regenerative and preventive medicine. As an example, a video analysis method for quantifying the rate of hESC colony growth was used to evaluate the toxicity of cigarette smoke from conventional and harm reduction cigarettes [25] [26]. The hESC were imaged over time using a high content Nikon BioStation IM incubation unit equipped with a phase contrast microscope. Time-lapse videos were evaluated quantitatively for colony growth during treatment with cigarette smoke. Analysis showed that side-stream smoke from “harm reduction” brands of cigarettes was as harmful as or even more harmful than side-stream smoke from a conventional brand [25].

Cell region detection using the BioStation’s cell analysis software is done either manually or in a semi-automatic manner [26]. The fastest rate at which BioStation IM can collect data is one frame per two seconds. In the current study, a new video

bioinformatics tool is developed to further enhance the analysis of hESC video data. With this new tool, cell regions are detected using a bio-driven algorithm that uses a mixture of two Gaussians and exploits properties of hESC. Once cell regions are detected, quantitative data can be utilized to determine the rate of hESC growth and numerous other parameters related to it such as its blebbing and attachment behavior. Therefore, high sensitivity and specificity on cell region detection are significant. Most importantly, the proposed method requires only 1.2 seconds of processing time per frame on a laptop with a Intel(R) Core™ 2 Duo CPU processor that run at 2.53GHz, it can perform cell analysis concurrently with the BioStation which is collecting live video data. The establishment of an automated and accurate cell detection tool is valuable and necessary for studying dynamic processes in hESCs.

### 3.1 Motivation

State of the art CL-Quant software [27] for bioinformatic image analysis requires users to make a recipe for the experimental data and the recipe is created with the data itself. It is semi-automatic, and its performance is heavily depended on the recipe maker. Our proposed method is intended to solve the connectivity problems by using cell property as well as the cell and substrate intensity distributions. Figure 1 show the connectivity problems in image for cell region detection: incomplete halo and similar substrate intensity values around the cell. The cell property manifests itself in spatial information where cell regions have a high intensity variation. This variation in cell region is due to the organelles inside the cell. We evolve the cell regions based on spatial

information until the optimal intensity distributions of background (substrate) and foreground (hESC) regions are obtained. The optimization is done on the original image and the spatial evolution is based on the spatial characteristic. The proposed method is bio-driven, fast and automated.

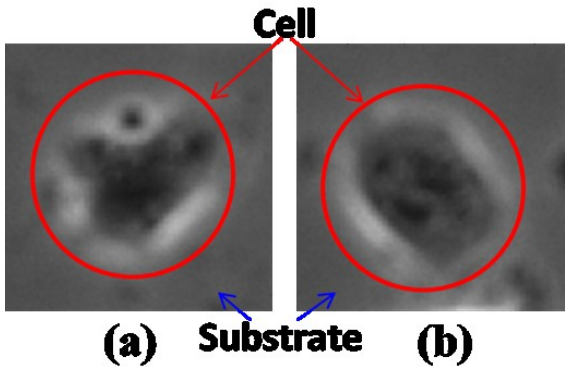



**Figure 1** (a)-(b). Cells with incomplete halo and similar substrate intensity values.

### 3.2 Technical Approach

In this subsection, we first explain the optimization metric modeled as a mixture of two Gaussians, and its convergence. We then elaborate on hESC property as spatial information. The handling of noise and over-segmentation are also discussed in this section. For the convenience of a reader, a summary of the symbols used in this chapter is provided in Table 1.

#### *3.2.1 Optimization Metric*

The hESC were cultured in vitro using methods described in detail previously [28]. The hESC are grown in culture dishes coated with a layer of substrate (Matrigel). The substrate becomes the background after the hESC are placed on its surface. Therefore, we model a hESC image with two regions of interest: foreground and

**Table 1 Definition of the Symbols Used in This Chapter**

| Symbol | Definition |
|---|---|
| $MVR$ | Mean-to-variance ratio |
| $M_i$ | Optimization metric $M$ at the $i_{th}$ iteration |
| F | Foreground/cell region |
| B | Background/substrate |
| $\sigma_f^2$ | Foreground intensity variance |
| $\sigma_b^2$ | Background intensity variance |
| $\mu_f$ | Foreground mean intensity value |
| $\mu_b$ | Background mean intensity value |
| G | Squared gradient magnitude of image |
| $I_G$ | Spatial information/high intensity variation due to organelles in the cell |
| $H$ | Normalized low pass Gaussian mask |
| $I_F$ | Filtered Image |

background. Figure 2 shows that the intensity distributions of these regions are similar to a mixture of two Gaussians with different means and variances. Consequently, we model the intensity distribution of foreground (cell region with a mean $\mu_f$ and variance $\sigma_f^2$) and background (substrate region with a mean $\mu_b$ and variance $\sigma_b^2$) as the mixture of two Gaussians. Figure 3 shows our model.

With this model, we then want to maximize the absolute difference of two mean-to-variance ratios (MVRs); the absolute difference of the foreground MVR and background MVR. The MVRs of the foreground and background datasets are calculated by the following equations [29].

$$\mathrm{MVR_f} = \frac{\mu_f}{\sigma_f^2} \tag{3.1}$$

$$\mathrm{MVR_b} = \frac{\mu_b}{\sigma_b^2} \tag{3.2}$$

where $\mathrm{MVR_f}$ and $\mathrm{MVR_b}$ are the MVRs for the foreground and background datasets respectively.

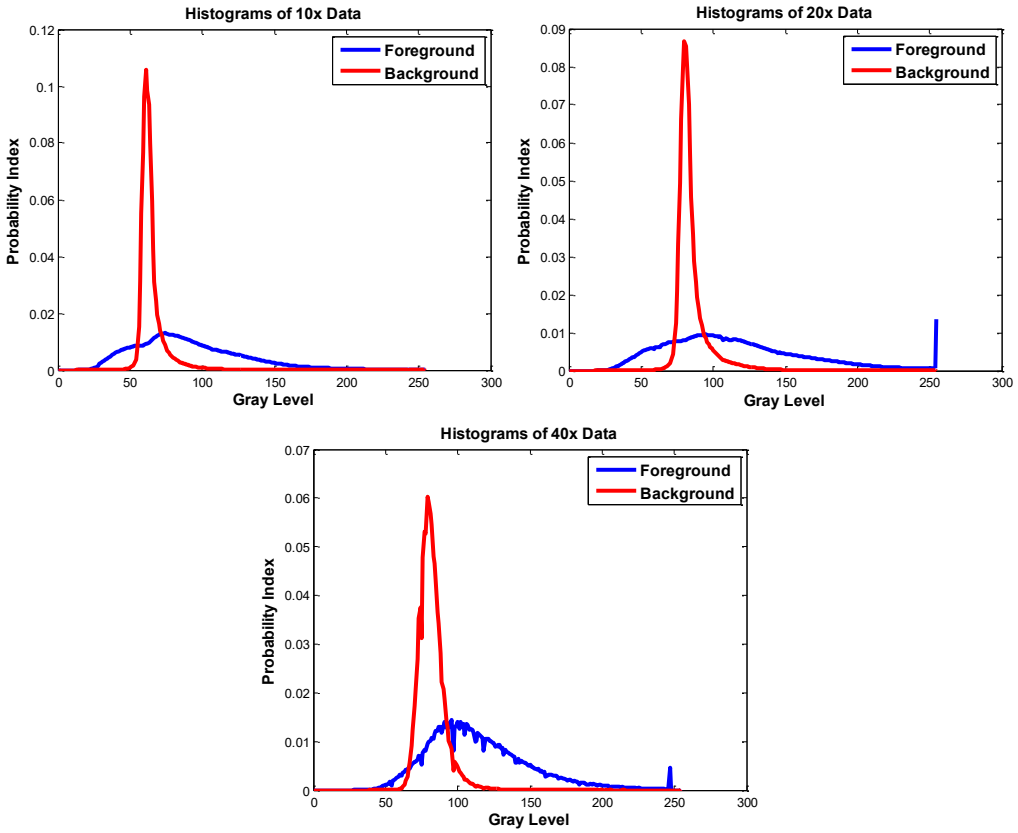



**Figure 2** Foreground and background intensity distributions for each data set.

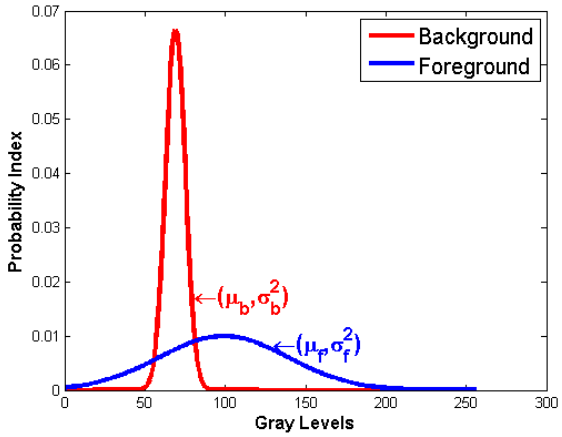



**Figure 3** Intensity distribution models of foreground and background.

The optimization metric M is formulated as:

$$M = |MVR_f - MVR_b| \tag{3.3}$$

Substituting (3.1) and (3.2) into (3.3), we get the following.

$$M = \left| \frac{\mu_f}{\sigma_f^2} - \frac{\mu_b}{\sigma_b^2} \right| \tag{3.4}$$

Equation (3.4) shows the metric that is used to determine how much the cell region data are different from the substrate region data. Since the algorithm is spatially evolving the foreground region from the initial high intensity variation region by a mean filter at each iteration, the foreground mean and variance are approaching to the background mean and variance. The limit of M is 0 as $\frac{\mu_f}{\sigma_f^2}$ approaches to $\frac{\mu_b}{\sigma_b^2}$. Therefore, our problem becomes finding $M_{opt}$which is the optimal value for metric M, and the corresponding equation is described below:

$$M_{opt} = \max_{\mu_f, \sigma_f^2, \mu_b, \sigma_b^2} M(\mu_f, \sigma_f^2, \mu_b, \sigma_b^2) \tag{3.5}$$

$M_{opt}$ finds the parameters that maximize the difference between foreground and background data.

*3.2.2 Convergence of the Metric*

The convergence of metric M can be proven from experiments. Figure 4 shows

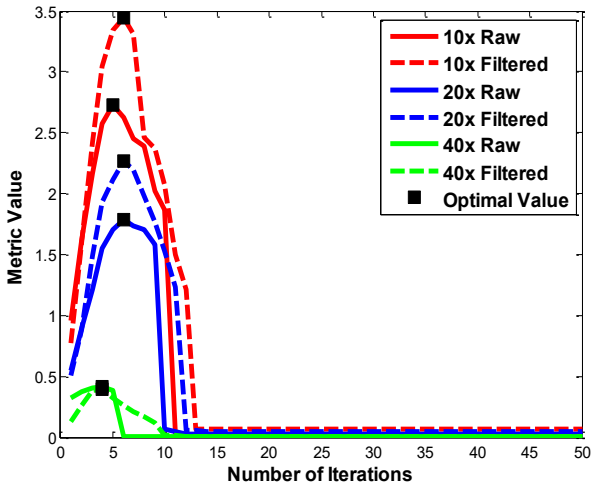



**Figure 4** Metric values at each iteration for images under different objectives.

the metric M at each iteration for all objectives with/without filtering.

*3.2.3 Spatial Information and Intensity Distribution*

The hESC region, F, is a high intensity variation region while the substrate region, B, is a low intensity variation region. As a result, we are able to exploit the gradients of the image to segment out the cell region from the substrate region. The following equations show how we exploit the gradients of the image:

$$I = F \cup B \tag{3.6}$$

$$G = \left(\frac{dI}{dx}\right)^2 + \left(\frac{dI}{dy}\right)^2 \tag{3.7}$$

$$I_G = \log_e\left(\frac{(-1+e)\times G}{\max(G)} + 1\right) \times 255 \tag{3.8}$$

where G is the squared gradient magnitude of image, I. $\frac{dI}{dx}$ and $\frac{dI}{dy}$ are gradients of image, I, in the x and y directions, respectively. $I_G$ is the spatial information produced by (3.8), which further emphasizes the difference between cell and substrate regions. Equation (3.8) normalizes G. The inner component of natural log transformation, $((-1 + e) \times G/\max(G)) + 1$, ensures that the transformation result will be within the range from 0 to 1. When G is 0, then $\log_e(1)$ is equal to 0. When G is equal to the max of G, then $\log_e(e)$ is equal to 1 and $I_G$ is equal to 255. The natural log function transforms a narrow range of small input values into a wider range of output values. Equation (3.8) is essentially a gamma correction technique [30]. It creates a large intensity separation between the foreground and background. Therefore, the natural log transformation enhances the image's intensity distribution to become a more visible bimodal distribution.

The proposed algorithm also uses a mean filter on $I_G$ at each iteration to evolve the cell regions. It is able to group the cell region pixels together based on local information; the size of the mean filter dictates how fast the cell region is evolved. The method updates $I_G$ and evolves the cell region until M is maximized.

Equation (3.4) is calculated based on the mean and variance of the intensity distributions of the cell and substrate data. The cell region, F, and substrate region, B, are updated by thresholding $I_G$ with OTSU's method at each iteration [30]. The intensity distribution's mean and variance of the cell region and substrate region data are also updated at each iteration by the following equations:

$$\mu_f = \frac{\sum_{f \in F} f}{N_f} \tag{3.9}$$

$$\mu_b = \frac{\sum_{b \in B} b}{N_b} \tag{3.10}$$

$$\sigma_f^2 = \frac{\sum_{f \in F} (f - \mu_f)^2}{N_f} \tag{3.11}$$

$$\sigma_b^2 = \frac{\sum_{b \in B} (b - \mu_b)^2}{N_b} \tag{3.12}$$

where $N_f$ and $N_b$ are the total numbers of foreground and background pixels in the image, f and b are the intensity values in the corresponding foreground and background. Figure 5 shows the intermediate and final results of the proposed method on various images. Figure 5(b), 5(e), 5(h) and 5(k) show the spatial information when $M_{opt}$ is reached for their respective data. The pseudo algorithm is given in Algorithm 1.

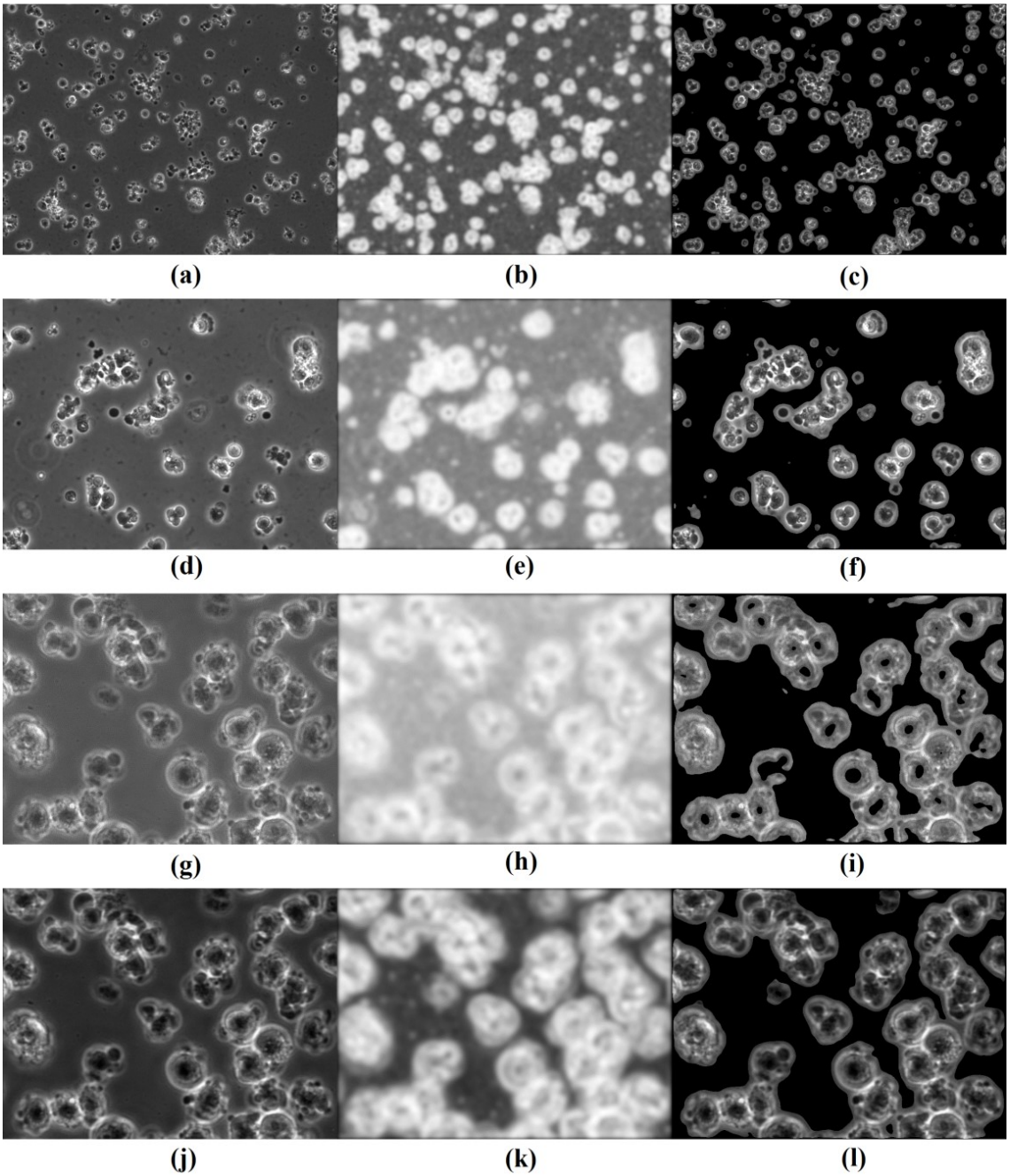



**Figure 5** (a) The original 10x image; (b) 10x image's spatial information at $\boldsymbol{M_{opt}}$; (c) detected cell regions in 10x image; (d) The original 20x image; (e) 20x image's spatial information at $\boldsymbol{M_{opt}}$;(f) detected cell regions in 20x image. (g) The original 40x image; (h) 40x image's spatial information at $\boldsymbol{M_{opt}}$; (i) detected cell regions in 40x image. (j) The 40x image after the low pass Gaussian filter; (k) The filtered 40x image's spatial information at $\boldsymbol{M_{opt}}$; (l) detected cell regions in the filtered 40x image.

**Algorithm 1: Cell Region Detection**

Input:
I: hESC phase contrast image.
Output:
F: the hESC region (foreground).
B: the substrate region (background).
1: **Procedure** Cell_Region_Detection(I);
2: Set $M_0 = 0$.
3: Set iteration $i \leftarrow 1$.
4: Calculate G and $I_G$ With equation (3.7) and (3.8).
5: Update $I_G$ by applying a mean filter to $I_G$ from step 3 (spatial grouping).
6: Determine foreground and background regions $F_1$ and $B_1$ respectively by applying Otsu's method to $I_G$.
7: Calculate $\mu_f$, $\mu_b$, $\sigma_f^2$ and $\sigma_b^2$ using equations (3.9)-(3.12).
8: Calculate $M_1$ with equation (3.4).
9: While($M_i > M_{i-1}$){
10: Iteration $i \leftarrow i + 1$.
11: Update $I_G$ by applying a mean filter to $I_G$ from the last iteration.
12: Determine $F_i$ and $B_i$ regions by applying Otsu's method to $I_G$.
13: Update $\mu_f$, $\mu_b$, $\sigma_f^2$ and $\sigma_b^2$ with equations (3.9)-(3.12).
14: Update $M_i$ with equation (3.4).
15: }
16: F $\leftarrow F_i$;
17: B $\leftarrow B_i$;
18: **end procedure**

*3.2.4 Noise Reduction*

In some cases, the effect of system noise from the microscope is inevitably visible. One way to reduce the effect of the noise in the image is to use a conventional filtering technique in the frequency domain. We use the low pass Gaussian mask to attenuate the noise in the image. This allows the improvement of detection accuracy. Figure 6 shows the process and results of the filtering with a low pass Gaussian mask, H, and the equations for noise reduction technique are given below [30].

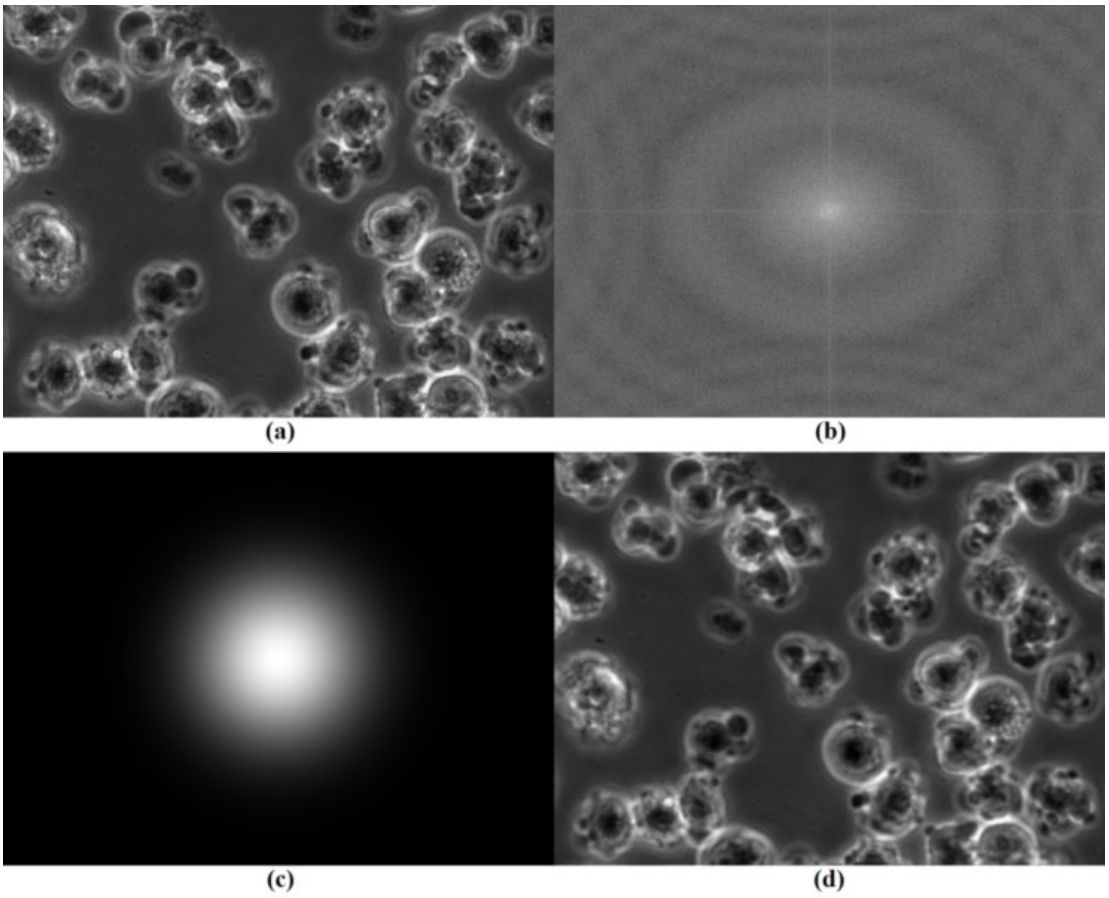



**Figure 6** (a) The original noisy 40x image; (b) noisy 40x image after 2D Fourier transformation; (c) low pass Gaussian mask with standard deviation equal to 80; (d) resulting image after noise filtering.

$$h_{gaus}(r,c) = e^{-\frac{r^2+c^2}{2\sigma_{gaus}^2}} \tag{3.13}$$

$$H(r,c) = \frac{h_{gaus}(r,c)}{\sum_r \sum_c h_{gaus}(r,c)} \tag{3.14}$$

$$I_F = \mathcal{F}^{-1}\{\mathcal{F}\{I\} \circ \mathcal{F}\{H\}\} \tag{3.15}$$

$I_F$ and H have the same dimensionality as image, I, where it has R rows and C columns. $r \in \{1,..,R\}$ and $c \in \{1,..,C\}$. $h_{gaus}(r,c)$ is a low pass Gaussian mask value at location $(r,c)$ and $\sigma_{gaus}$ is the standard deviation of the Gaussian mask. $\mathcal{F}\{*\}$ is a 2D Fourier transform operation, and $\mathcal{F}^{-1}\{*\}$ is the inverse 2D Fourier transform operation.

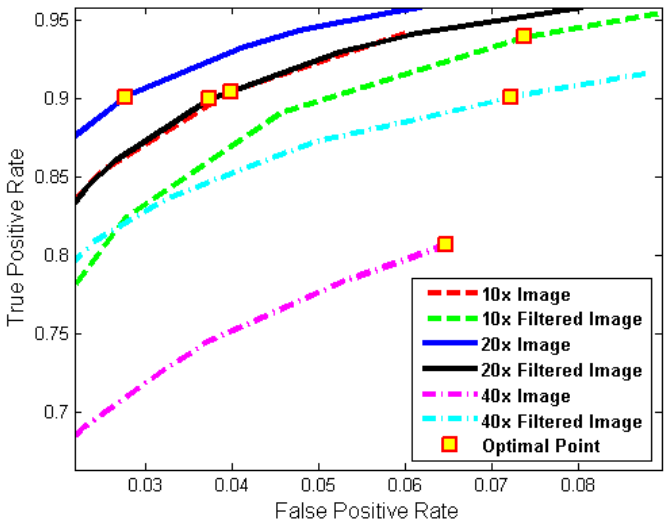



**Figure 7** ROC plots for images under different objectives with varying erosion parameter.

*3.2.5 Over-Segmentation Reduction*

Since the mean filter is used to evolve the foreground region, over-segmentation is inevitable. Therefore, we use a morphological erosion technique to reduce the error caused by over-segmentation. The erosion parameter is identified from receiver operating characteristic (ROC) curve where the minimum of 90% true positive rate and a false positive rate lower than 10% are achieved for each data set with/without filtering [31]. Figure 7 shows the ROC and the optimal points where the erosion parameters are picked.

## 3.3 Experimental Results

*3.3.1 Data*

All time lapse videos were obtained with a BioStation IM [32]. The frames in the video are phase contrast images with 600 x 800 resolutions. The videos were acquired using three different objectives: 10x, 20x and 40x and each objective has a set of 40

images, with a total of 120 images. Each video frame is taken roughly 2 minutes apart for the purpose of data variation from frame to frame. The ground-truth is generated manually by the expert biologists.

Note that for all the video data used in this paper, the hESC culture conditions are considered to be excellent. All videos used are of small colonies or single hESC, and the cells look excellent for unattached hESC and colonies. Most hESC culture today is not done on mouse embryonic fibroblasts (MEFs). We have not cultured hESC on feeders since 2008. We use mTeSR medium [33]. This modern culture media does not require the use of MEFs, so they are seldom used. With an exception for maintenance, MEFs are not used in experiments. Since it is highly unlikely that we will analyze hESC cultured on MEFs, we have not tested our algorithms on data sets with hESC cultured on MEFs. Moreover, the images in this manuscript have very few dead cells and debris. In fact, they are remarkably clean considering the cells have been stripped and replated.

*3.3.2 Parameters*

Each video collected with different objectives has a different default size of neighborhood for spatial grouping. The default sizes are determined by observing the ROC plots with various window sizes for each objective. Based on the experimental analysis in [10], we concluded that the optimal neighborhood sizes for 10x, 20x and 40x are 5 x 5, 7 x 7 and 11 x 11 respectively. The selection criteria for the neighborhood sizes are based on finding a window for which its ROC plot yields a high true positive rate while keeping the low false positive rate. A low pass Gaussian mask with a standard deviation equal to 80 pixels is used to get rid of the noise that occurs during the video

acquisition process. For erosion parameters, we use a disk with radius 2, 5, 4, 8, 0, and 1 for 10x, filtered 10x, 20x, filtered 20x, 40x and filtered 40x datasets, respectively.

*3.3.3 Performance Measures*

The true positive, TP, is the overlapped region between the detected cell region and the cell region ground-truth. True negative, TN, is the overlapped region between the detected background region and the background ground-truth. The false positive, FP, is the detected background that is falsely identified as part of the cell region. The false negative, FN, is the detected cell region that is falsely identified as part of the background.

The true positive rate or sensitivity, TPR or SEN, measures the proportion of actual positives which are correctly identified.

$$\mathrm{TPR} = \frac{\mathrm{TP}}{(\mathrm{TP}+\mathrm{FN})} \tag{3.16}$$

The false positive rate, FPR, measures the proportion of false positives which are incorrectly identified.

$$\mathrm{FPR} = \frac{\mathrm{FP}}{(\mathrm{FP}+\mathrm{TN})} \tag{3.17}$$

The specificity, SPC, is the true negative rate which is a complement of false positive rate.

$$\mathrm{SPC} = \frac{\mathrm{TN}}{(\mathrm{FP}+\mathrm{TN})} \tag{3.18}$$

The Jaccard similarity, JAC, is a measure of similarity between experimental results and the ground-truth.

$$\mathrm{JAC} = \frac{\mathrm{TP}}{(\mathrm{TP}+\mathrm{FP}+\mathrm{FN})} \tag{3.19}$$

The Dice coefficient, DIC, measures the agreement between experimental results and ground-truth.

$$\mathrm{DIC} = \frac{2\mathrm{TP}}{(2\mathrm{TP}+\mathrm{FP}+\mathrm{FN})} \tag{3.20}$$

The average detection error is an average of type I (1-SPC) and type II (1-SEN). ANOVA test [34] is also used for comparison of detected foreground and background intensity distributions with the corresponding ground-truth intensity distributions.

### *3.3.4 Methods Compared*

We compared the proposed method with k-means, Mixture of Gaussians, and CL-Quant software with various recipes [4] [7] [27]. In addition, we evaluated the datasets with Otsu's algorithm [30]. However, it was not able to detect the entire cell region due to fact that the intensity values of the cell body are similar to the substrate intensity values. As shown in Figure 8, the result was not useful. Therefore, Otsu's algorithm is not compared in this paper.

### *3.3.5 Results and Discussion*

The proposed method was tested with three videos (each with 40 frames) that were acquired with 10x, 20x and 40x objectives. The proposed method achieves above 90% in sensitivity and specificity on 10x with/without filtering, 20x with/without filtering and 40x with filtering datasets. Since pre-filtering gets rid of high frequency noise, it improves the performance of the proposed algorithm on noisy data. The 10x and 20x datasets are not corrupted with high frequency noise. Therefore, pre-filtering on those datasets would not affect the algorithm's performance on JAC, DIC, SEN and SPC

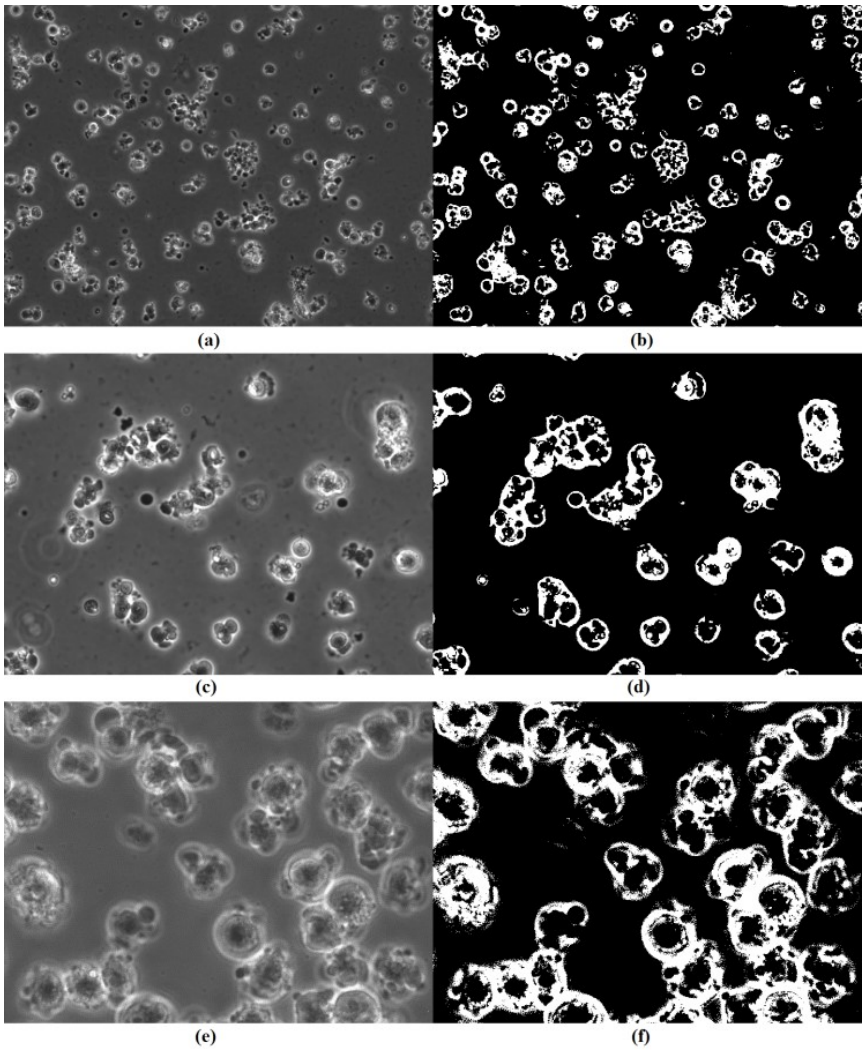



**Figure 8** (a) The original 10x image; (b) binary result of (a) with Otsu's; (c) the original 20x image; (d) binary result of (c) with Otsu's; (e) the original 40x image; (f) binary result of (e) with Otsu's.

measures as shown in Table 2 and 3. However, the 40x dataset is corrupted with high frequency noise. The pre-filtering improves the yield on its SEN measure significantly as shown in Table 4. In this paper, we compared the proposed method with K-means, mixture of Gaussians segmentation method and CL-Quant software under different recipes. Tables 2-4 show the results of the K-means, MGEM segmentation, CL-Quant software, and the proposed method on all experimental data. The proposed method

**Table 2 Comparisons of 10x Data Set (* denotes filtered data)**

| Method | Jaccard | Dice | SEN | SPC |
|---|---|---|---|---|
| Proposed | 0.737 | 0.848 | 0.907 | 0.911 |
| Proposed* | 0.728 | 0.843 | 0.905 | 0.907 |
| KM | 0.465 | 0.635 | 0.519 | 0.956 |
| KM* | 0.403 | 0.574 | 0.428 | 0.977 |
| MGEM | 0.608 | 0.756 | 0.793 | 0.884 |
| MGEM* | 0.591 | 0.743 | 0.852 | 0.831 |
| CLQuant with 10x recipe | 0.674 | 0.806 | 0.722 | 0.973 |
| CLQuant* with 10x recipe | 0.579 | 0.734 | 0.610 | 0.980 |
| CLQuant with 20x recipe | 0.622 | 0.767 | 0.978 | 0.781 |
| CLQuant* with 20x recipe | 0.538 | 0.700 | 0.988 | 0.680 |
| CLQuant with 40x recipe | 0.558 | 0.716 | 0.918 | 0.753 |
| CLQuant* with 40x recipe | 0.477 | 0.646 | 0.916 | 0.648 |

**Table 3 Comparisons of 20x Data Set (* denotes filtered data)**

| Method | Jaccard | Dice | SEN | SPC |
|---|---|---|---|---|
| Proposed | 0.754 | 0.860 | 0.906 | 0.924 |
| Proposed* | 0.739 | 0.850 | 0.908 | 0.914 |
| KM | 0.446 | 0.617 | 0.516 | 0.941 |
| KM* | 0.358 | 0.527 | 0.378 | 0.978 |
| MGEM | 0.585 | 0.738 | 0.804 | 0.859 |
| MGEM* | 0.581 | 0.735 | 0.827 | 0.840 |
| CLQuant with 10x recipe | 0.689 | 0.816 | 0.767 | 0.958 |
| CLQuant* with 10x recipe | 0.652 | 0.789 | 0.709 | 0.967 |
| CLQuant with 20x recipe | 0.663 | 0.797 | 0.797 | 0.924 |
| CLQuant* with 20x recipe | 0.600 | 0.750 | 0.860 | 0.837 |
| CLQuant with 40x recipe | 0.596 | 0.747 | 0.915 | 0.799 |
| CLQuant* with 40x recipe | 0.547 | 0.707 | 0.932 | 0.735 |

outperforms the other methods in JAC and DIC measures. Moreover, the proposed method yields above 90% in both SEN and SPC measures. K-means clusters the image data based only on the nearest mean while the MGEM method groups the data solely on the modeled intensity distribution. Consequently, neither method was able to detect the entire cell regions. Instead, they have detected fragments of the actual cell regions. Their

**Table 4 Comparisons of 40x Data Set (* denotes filtered data)**

| Method | Jaccard | Dice | SEN | SPC |
|---|---|---|---|---|
| Proposed | 0.735 | 0.847 | 0.749 | 0.968 |
| Proposed* | 0.866 | 0.928 | 0.902 | 0.931 |
| KM | 0.674 | 0.805 | 0.797 | 0.697 |
| KM* | 0.411 | 0.583 | 0.412 | 0.995 |
| MGEM | 0.621 | 0.766 | 0.634 | 0.963 |
| MGEM* | 0.775 | 0.874 | 0.827 | 0.890 |
| CLQuant with 10x recipe | 0.724 | 0.840 | 0.734 | 0.976 |
| CLQuant* with 10x recipe | 0.567 | 0.724 | 0.569 | 0.994 |
| CLQuant with 20x recipe | 0.792 | 0.884 | 0.885 | 0.803 |
| CLQuant* with 20x recipe | 0.737 | 0.849 | 0.801 | 0.855 |
| CLQuant with 40x recipe | 0.831 | 0.908 | 0.952 | 0.758 |
| CLQuant* with 40x recipe | 0.805 | 0.892 | 0.930 | 0.741 |

**Table 5 Average Detection Errors of Foreground and Background**

| Data | K-means | MGEM | CL Quant with 10x Recipe | CL Quant with 20x Recipe | CL Quant with 40x Recipe | Proposed |
|---|---|---|---|---|---|---|
| 10x Data | 26.28% | 16.18% | 15.26% | 12.05% | 16.46% | **9.07%** |
| 10x Data (Filtered) | 29.79% | 15.84% | 20.51% | 16.61% | 21.80% | **9.41%** |
| 20x Data | 27.15% | 16.83% | 13.77% | 13.96% | 14.34% | **8.47%** |
| 20x Data (Filtered) | 32.16% | 16.62% | 16.20% | 15.16% | 16.62% | **8.90%** |
| 40x Data | 25.33% | 20.14% | 14.46% | 15.60% | 14.50% | **14.17%** |
| 40x Data (Filtered) | 29.64% | 14.17% | 21.84% | 17.20% | 16.46% | **8.36%** |

performance is further degraded by the presence of noise. The CL-Quant software's performance depends heavily on the recipe maker. The recipes for the datasets used in this paper are created by a fourth year biology Ph.D. student. More importantly, the proposed method's performance is good on the image data with or without filtering.

In term of performance, the proposed method yields lower than 10% average detection error of foreground and background on 10x and 20x with/without filtering, and

40x with filtering datasets as shown in Table 5. MGEM has a minimum of 14.17% and a maximum of 20.14% average detection error [7]. K-means algorithm yields above 25% average detection error on all datasets [4]. CL-Quant gives a minimum of 12.05% and a maximum of 21.84% average detection error [27]. In terms of convergence, the proposed method converges in 7 iterations on the average, and each iteration requires 0.17 second. It reaches the global optimum since the mean filter is used for grouping similar regions in the algorithm.

### 3.4 Conclusion

The proposed method incorporated the concept of local property of a hESC as well as cell and substrate intensity distributions for cell region detection in phase contrast images. It uses the spatial information to improve the connectivity of local pixels to their corresponding regions. More importantly, it enables fast convergence to the maximum absolute difference of foreground and background mean-to-variance ratios. The proposed method is able to split the image data into two Gaussian distributions; intensity distribution of the foreground and background data. Table 5 shows that the proposed method yields a lower average detection error than the K-means, MGEM and CL-Quant methods [4] [7] [27]. Figure 9 shows an ANOVA test for all experimental data sets. It shows low error in comparison between the intensity distributions of the proposed method and the ground-truth intensity distributions. In the case of noisy images, the pre-filtering of the image data can greatly improve the performance of the algorithm. In term of speed, the proposed method converges in less than 1.2 seconds while K-means and MGEM take about 3.61 and 25.3 seconds respectively on a laptop with a Intel(R) Core™

2 Duo CPU processor that run at 2.53GHz. The CL-Quant software requires at least 6 minutes of user inputs from the expert biologist for each recipe. Application of this automated method to hESC will facilitate the analysis of their dynamic behaviors and benefit research in both regenerative and preventive medicine.

**Anova Test on 10X's Data Results by the Proposed Method without Filtering**

**Foreground Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 3.02E+11 | 255 | 1.19E+09 | 37.25 | 0 |
| 'Error' | 8.15E+09 | 256 | 3.18E+07 | [] | [] |
| 'Total' | 3.10E+11 | 511 | [] | [] | [] |

**Background Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 2.30E+13 | 255 | 9.01E+10 | 2831.21 | 0 |
| 'Error' | 8.15E+09 | 256 | 3.18E+07 | [] | [] |
| 'Total' | 2.30E+13 | 511 | [] | [] | [] |

**Anova Test on 10X's Data Results by the Proposed Method with Filtering**

**Foreground Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 3.11E+11 | 255 | 1.22E+09 | 32.07 | 0 |
| 'Error' | 9.73E+09 | 256 | 3.80E+07 | [] | [] |
| 'Total' | 3.21E+11 | 511 | [] | [] | [] |

**Background Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 2.30E+13 | 255 | 9.00E+10 | 2368.96 | 0 |
| 'Error' | 9.73E+09 | 256 | 3.80E+07 | [] | [] |
| 'Total' | 2.30E+13 | 511 | [] | [] | [] |

**Anova Test on 20X's Data Results by the Proposed Method without Filtering**

**Foreground Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 1.84E+11 | 255 | 7.20E+08 | 32.75 | 0 |
| 'Error' | 5.63E+09 | 256 | 2.20E+07 | [] | [] |
| 'Total' | 1.89E+11 | 511 | [] | [] | [] |

**Background Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 1.83E+13 | 255 | 7.16E+10 | 3255.54 | 0 |
| 'Error' | 5.63E+09 | 256 | 2.20E+07 | [] | [] |
| 'Total' | 1.83E+13 | 511 | [] | [] | [] |

**Anova Test on 20X's Data Results by the Proposed Method with Filtering**

**Foreground Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 1.93E+11 | 255 | 7.56E+08 | 27.16 | 0 |
| 'Error' | 7.13E+09 | 256 | 2.78E+07 | [] | [] |
| 'Total' | 2.00E+11 | 511 | [] | [] | [] |

**Background Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 1.82E+13 | 255 | 7.13E+10 | 2562.46 | 0 |
| 'Error' | 7.13E+09 | 256 | 2.78E+07 | [] | [] |
| 'Total' | 1.82E+13 | 511 | [] | [] | [] |

**Anova Test on 40X's Data Results by the Proposed Method without Filtering**

**Foreground Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 1.10E+12 | 255 | 4.30E+09 | 18.21 | 0 |
| 'Error' | 6.04E+10 | 256 | 2.36E+08 | [] | [] |
| 'Total' | 1.16E+12 | 511 | [] | [] | [] |

**Background Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 4.22E+12 | 255 | 1.65E+10 | 70.10 | 0 |
| 'Error' | 6.04E+10 | 256 | 2.36E+08 | [] | [] |
| 'Total' | 4.28E+12 | 511 | [] | [] | [] |

**Anova Test on 40X's Data Results by the Proposed Method with Filtering**

**Foreground Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 1.33E+12 | 255 | 5.23E+09 | 253.49 | 0 |
| 'Error' | 5.28E+09 | 256 | 2.06E+07 | [] | [] |
| 'Total' | 1.34E+12 | 511 | [] | [] | [] |

**Background Distribution**

| 'Source' | 'SS' | 'df' | 'MS' | 'F' | 'Prob>F' |
|---|---|---|---|---|---|
| 'Columns' | 3.77E+12 | 255 | 1.48E+10 | 716.63 | 0 |
| 'Error' | 5.28E+09 | 256 | 2.06E+07 | [] | [] |
| 'Total' | 3.78E+12 | 511 | [] | [] | [] |

**Figure 9** ANOVA test of foreground and background distributions for all data sets. SS = sum of squares, df = degree of freedom, MS = mean square, F = F-statistic, [] = not applicable.

# CHAPTER 4 Bio-Optimized Segmentation of Blebs in Human Embryonic Stem Cell Videos

The detection of blebs is important for understanding human embryonic stem cell (hESC) blebbing behavior and can be used to evaluate cell health. The ability to analyze rates of bleb formation and retraction are important in the field of toxicology and could form the basis of an assay that depends on a functional cytoskeleton [24] [25] [35].

The experiment in this chapter uses the H9 line of hESC (WiCell, Madison, WI.) which are normally about 10 microns in diameter. The average bleb-to-cell body ratio is about 16:57. In 2D images, a single cell can have an average of 6 or a maximum of 11 blebs with a 20x objective. This chapter is intended to explain the bleb formation phenomena and to introduce a method to segment out sequences of bleb regions in video for further analysis. Bleb formation consists of two processes: 1) expansion; 2) retraction [11] [12]. During the first stage, the bleb expands sporadically. During the retraction stage, the bleb either retracts back and disappears or partially retracts. Complete bleb retraction normally occurs during dynamic blebbing, which is characteristic of a healthy cells. However, if bleb retraction does not occur or occurs slowly, the cell is likely undergoing apoptosis or cell death. Figure 10 shows the sequences of both bleb expansion and retraction. During expansion, bleb size increases while cell body size decreases. In contrast, during bleb retraction, bleb size decreases while cell body size increases. At the final stage of expansion, the bleb is called an intermediate bleb. The intermediate bleb indicates that transition from expansion to retraction is occurring. The intermediate bleb has the maximum bleb size during expansion.

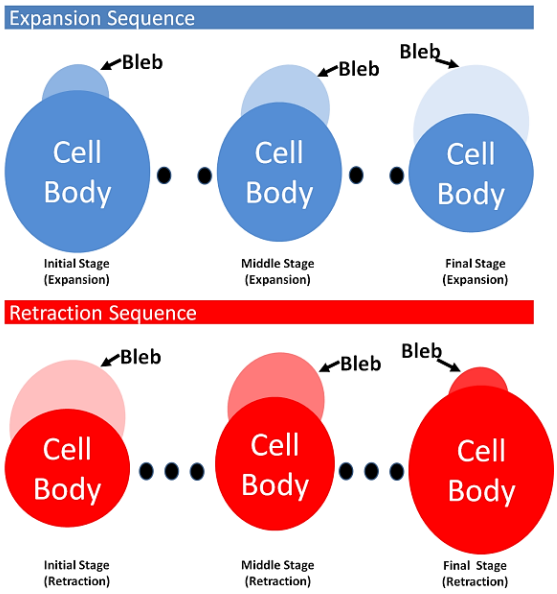



**Figure 10** Expansion and retraction processes occurring over time.

Figures 11 and 12 show phase contrast images of blebbing sequences for both expansion and retraction. The expansion and retraction processes are visually similar but biologically distinct [13] [14]. Computationally, we can consider one as a reversed sequence of the other yet with different rates in bleb size changes over time. Therefore, the retraction process can be viewed as a bleb in the final bleb expansion stage that retracts back to its initial bleb expansion stage. Because both processes are very different in their rates of bleb area change over time, we propose a bio-optimized approach for

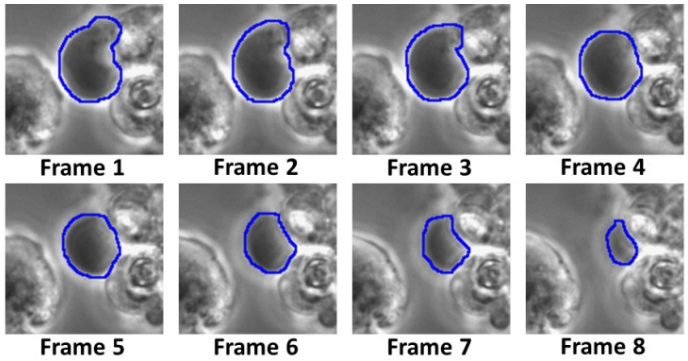



**Figure 11** Example of reversed bleb expansion sequence.

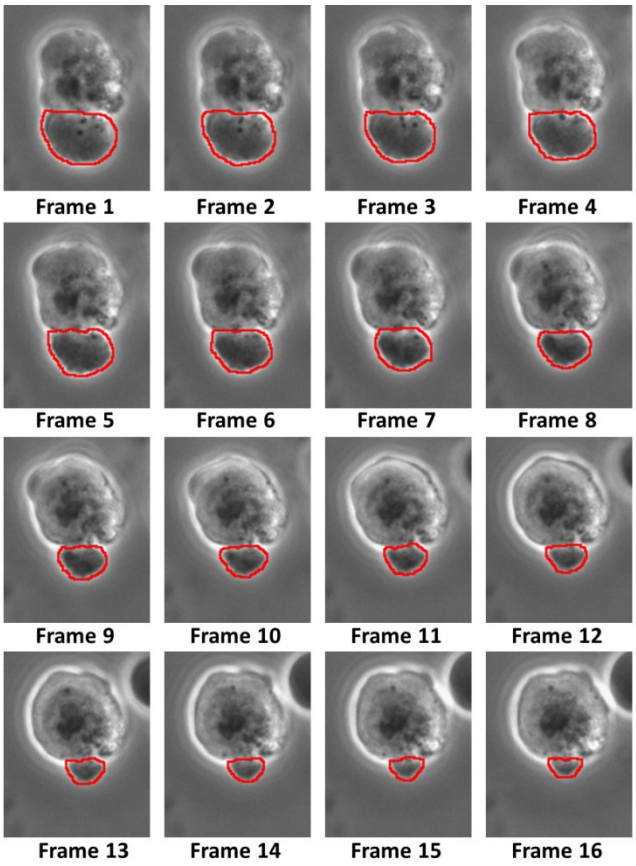



**Figure 12** Example of bleb retraction sequence.

image segmentations with two distinct models in this chapter. The expansion and retraction processes' area change are both characterized by a power law function but with different parameters.

Section 4.1 presents the motivation of this chapter. Section 4.2 describes the technical approach in detail. Section 4.3 provides experimental results and discussions on video data. Finally, Section 4.4 provides the conclusions of the chapter.

### 4.1 Motivation

Segmentation and object detection are closely related. Guan et al. [36] [10] present detection/segmentation methods for hESC in phase contrast images. Their

methods work for extraction of cell and cell colony from a phase contrast video. In this paper, we are concerned with the detection of blebs of a single cell. Since there is no previous work on detection of blebs, we are going to exploit the following methods.

Four commonly used methods for image segmentation are region growing, normalized cut, meanshift and watershed. Region growing (RG) is an attractive method for object detection. The seeded region growing method by Adams et al. [37] [38] grows a region initially from a seed point and groups its neighboring pixels to its region based on their similarity coefficient threshold. Its performance is dictated by the similarity threshold. The normalized cut (NC) method by Shi et al. [39] is a graph based approach. This approach performs segmentation by maximizing association within groups while minimizing disassociation between groups. This method requires prior knowledge on the number of segmented regions and performance depends on this parameter. The meanshift (MS) method by Comaniciu et al. [40] is a well-known density based approach. It partitions the image by assigning pixels into clusters with the same mode. The MS method requires spatial and range bandwidths as well as minimum region area as input, and its performance varies on those input parameters [41] [42]. The watershed (WS) method performs a flooding process on a gradient image where it starts at the local minima and builds watersheds to separate adjacent catchment basins [43]. The gradient image is generated by performing a Euclidean distance transform of the marker image. The marker image, which is a binary image, is obtained by finding the extended minimum of the original image [44]. Since the suppressing value of the extended minimum method needs to be known for the binarization of the gray level image, the

construction of the gradient image is also dependent on the suppressing value. Therefore, the right choice of a suppressing value is important for the watershed method.

Since a bleb is part of the cell, it brings the following challenges: i) bleb intensity and texture vary for different cells; ii) blebs are connected to the cell body; iii) blebs have similar intensity/texture as the cell body or background; iv) neighboring blebs share similar intensity and texture. Figure 13 shows a set of expanding and retracting blebs. Although these blebs look similar, they are different from each other in intensity and texture. Therefore, the conventional segmentation methods with constant parameters will not work well on all the bleb images in a video. Most importantly, blebbing is a dynamic process, and the bleb properties change over time. As a result, subsequent blebs in the same video sequence might have different image properties. Consequently, the performance of RG, NC, MS and watershed methods will suffer from any constant input parameters.

We proposed an interactive system with a bio-optimized segmentation method. The proposed approach adapts parameters for images in the bleb formation process. The parameters for the bio-optimized metric are derived directly from the expansion and retraction rates of bleb area change over time. Since the health status of a cell can be

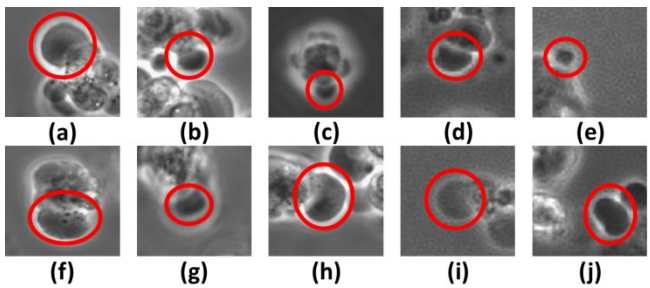



**Figure 13** (a)-(e) A sample of expanding blebs; (f)-(j) A sample of retracting blebs. (Note: the variety of blebs that are circled in these images)

determined from the bleb expansion and retraction time, it is possible to use the bleb formation process as a health index. Therefore, the proposed method is essential in developing an automated system for bleb analysis.

### 4.2 Technical Approach

In this section, we first introduce the concept of intermediate blebs in bleb detection. We then explain the derivation of the statistical models for the bio-optimized metric which is a part of an optimized segmentation approach. We also elaborate on the optimization metric. In addition, a flowchart of the bio-optimized segmentation approach is also provided. Figure 14 shows the overview of the proposed system. For the convenience of the reader, a summary of the symbols used in this paper is given in Table 6.

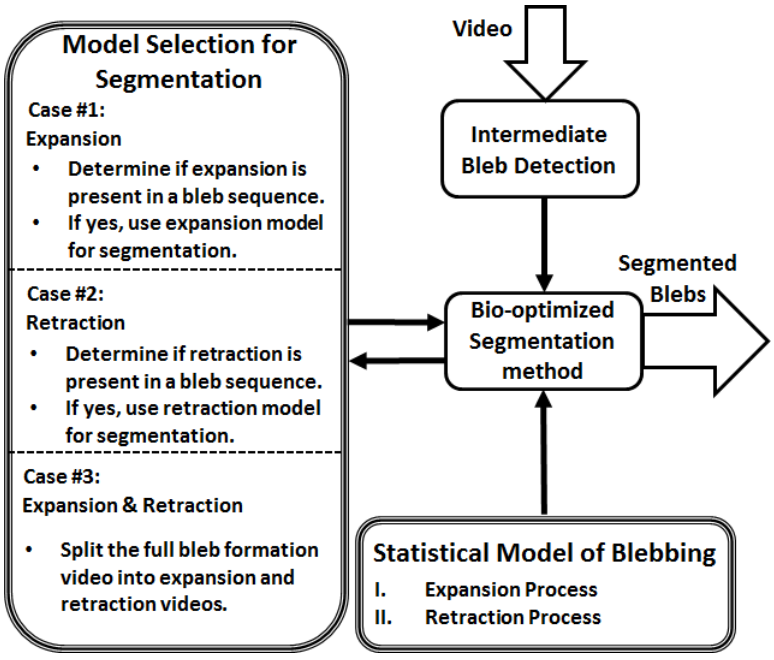



**Figure 14** Overview of the proposed system.

**Table 6 Definition of the Symbols Used in Chapter 4**

| Symbol | Definition |
|---|---|
| $t$ | Frame index value. |
| $E(t; A_E, B_E)$ | Exponential function. |
| $A_E$ | Overall scaling parameter of the exponential. |
| $B_E$ | Growth/Decay parameter of the exponential. |
| $L(t; A_L, B_L)$ | Linear function. |
| $A_L$ | The slope of the linear function. |
| $B_L$ | The y-intercept of the linear function |
| $G(t; A_G, B_G)$ | Gaussian function. |
| $A_G$ | The mean for the Gaussian function. |
| $B_G$ | The standard deviation for the Gaussian function. |
| $LOG(t; A_{\mathcal{N}}, B_{\mathcal{N}})$ | Logarithmic function. |
| $A_{Log}$ | Overall scaling parameter of the logarithmic. |
| $B_{Log}$ | The growth rate of the logarithmic function. |
| $\mathcal{P}(t; A_{\mathcal{P}}, B_{\mathcal{P}})$ | Power law function. |
| $A_{\mathcal{P}}$ | Overall scaling parameter of power law. |
| $B_{\mathcal{P}}$ | Power law exponent parameter. |
| $\Psi$ | Bio-optimization metric. |
| $Area_{init}$ | Initial intermediate bleb area. |
| $Area^t(i)$ | Area at $i_{th}$ iteration in frame index $t$. |
| $Area_{pre}(t)$ | Predicted area at index $t$ . |

*4.2.1 Intermediate Bleb Detection*

There are two important reasons for the detection of intermediate bleb. First, the intermediate bleb is a transitional bleb that divides expansion and retraction. Second, it has the largest bleb size in the bleb formation process. The significantly larger bleb size makes the detection of the intermediate bleb in the video easier. Since the intermediate bleb is the initial stage of retraction and the final stage of expansion, it is a good starting point for the bio-optimized segmentation approaches. Once the intermediate bleb frame is estimated, the video is then split into expansion and retraction processes. A decision

mechanism is used to determine which of the three cases, as shown in Figure 14, is occurring. The decision mechanism is further explained in later section for identifying whether an expansion or retraction process is occurring.

In this chapter, the intermediate blebs have been already identified by an expert biologist for both expansion and retraction videos. In the case of the full bleb formation videos, intermediate blebs were also identified by an expert biologist, and the results obtained are compared with the result from estimated intermediate bleb frame location method (see Section 4.3.5).

### *4.2.2 Statistical Model of Blebbing*

We used a model fitting technique on the average interpolated bleb area distributions to determine the best distribution to represent expansion and retraction. Figure 15 shows the model fitting with various distributions on the average interpolated sample area distribution data for the expansion and retraction processes. The average interpolated expansion and retraction sample area distributions are normalized area curves derived from the 29 sets of full bleb area distributions. The parameters for each curve in Figure 15 are shown in Table 7.

Figure 16 shows the distribution of both expansion and retraction training datasets derived from the 29 sets of full bleb formation datasets. Since the length of both expansion and retraction processes depends on the size of the intermediate bleb, the 29 sets of training data have different numbers of frames for each of the expansion and retraction area distribution curves. However, we assumed that the rate of bleb area

change is similar within expansion and retraction processes. Equations (4.1)-(4.5) are used in our model fitting technique [45].

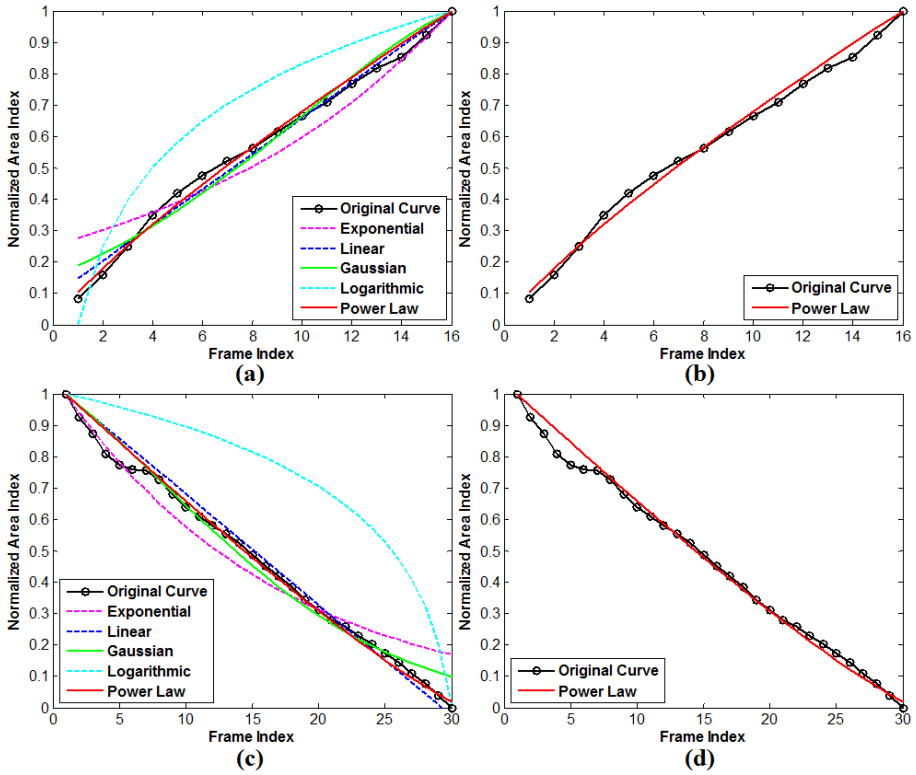



**Figure 15** Plots of curve fitting the average interpolated bleb area distribution: (a) in expansion for all methods; (b) in expansion for original and optimal curve. (c) in retraction for all methods; (d) in retraction for original and optimal curve.

Exponential Function:

$$E(t; A_E, B_E) = A_E e^{B_E t} \tag{4.1}$$

Linear Function:

$$L(t; A_L, B_L) = A_L t + B_L \tag{4.2}$$

Gaussian Function:

$$G(t; A_G, B_G) = \frac{1}{B_G\sqrt{2\pi}} e^{-\frac{(t-A_G)^2}{2B_G^2}} \tag{4.3}$$

**Table 7 Parameters for each function shown in Figure 15**

| Method | | Exp. | Linear | Gauss. | Log. | Power Law |
|---|---|---|---|---|---|---|
| **Expansion** | A | 6.377 | 1.870 | 20.077 | 0.000 | 4.192 |
| | B | 0.086 | 2.953 | 10.208 | 3.767 | 0.822 |
| **Retraction** | A | 5.939 | -0.135 | -10.226 | 0.000 | 5.472 |
| | B | 0.061 | 3.956 | 17.948 | 3.767 | 1.168 |

Note: Exp. → Exponential; Gauss. → Gaussian; Log. → Logarithmic;

**Table 8**
**Mean Squared error of Curve Fitting results for bleb area distribution**

| | | **Retraction** | | | | |
|---|---|---|---|---|---|---|
| | **Method** | **Exp.** | **Linear** | **Gauss.** | **Log.** | **Power Law** |
| **Expansion** | **Exp.** | 0.2235 | 0.1299 | 0.1376 | 2.5128 | 0.1131 |
| | **Linear** | 0.1468 | 0.0532 | 0.0609 | 2.4361 | 0.0364 |
| | **Gauss.** | 0.1650 | 0.0714 | 0.0791 | 2.4543 | 0.0546 |
| | **Log.** | 0.4392 | 0.3455 | 0.3533 | 2.7285 | 0.3287 |
| | **Power Law** | 0.1410 | 0.0474 | 0.0552 | 2.4303 | **0.0306** |

Note: Exp. → Exponential; Gauss. → Gaussian; Log. → Logarithmic;
(Bold number denotes the best performance that was achieved.)

Logarithmic Function:

$$\mathrm{LOG}(t; A_{Log}, B_{Log}) = A_{Log} + B_{Log} \ln(t) \tag{4.4}$$

Power Law Function:

$$\mathcal{P}(t; A_{\mathcal{P}}, B_{\mathcal{P}}) = A_{\mathcal{P}} t^{B_{\mathcal{P}}} \tag{4.5}$$

The variable $t$ is the frame index value which varies from 1 to 16 (longest length) for expansion and from 1 to 30 (longest length) for retraction. The statistics in Table 8 shows the total mean squared errors (MSE) of each combination of fittings during expansion and retraction processes. Since the power law function has the low average

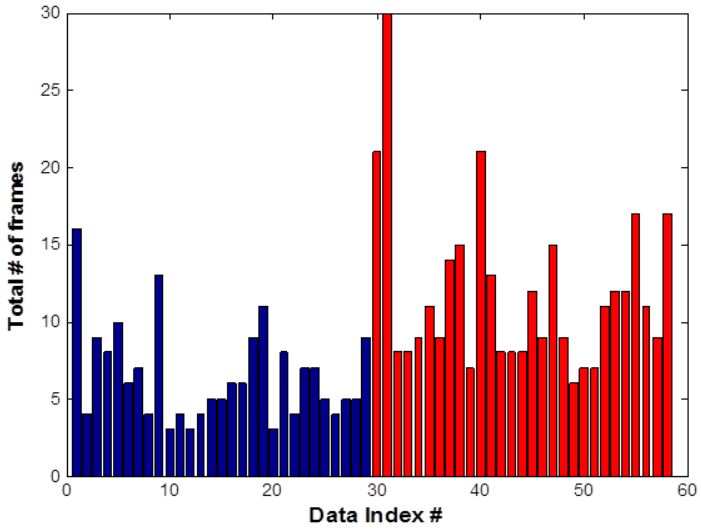



**Figure 16** Frame distribution of training dataset for expansion (blue) and retraction (red).

MSE for both expansion and retraction processes, we conclude that expansion and retraction area distributions can be represented by equation (4.5) with different parameters.

### *4.2.3 Bio-optimized Segmentation*

#### *4.2.3.1 Usage of Bio-optimization Metric in Segmentation Methods*

The bio-optimization metric provides an adaptive solution to the segmentation problem. It finds parameters in segmentation methods that yield the best solution to a given predicted area distribution in the video sequence.

Region growing is a region based segmentation method. It starts with an initial seed point and iteratively evolves its region by evaluating its region's neighboring contour. It groups the contour pixels based on a similarity threshold. The contour pixels are grouped into the region if the similarity between the pixel and region feature is less than a threshold. As a result, the performance of the region growing method depends on

the selection of the threshold. In the bio-optimized region growing approach, the threshold is an adaptive parameter that is needed to be found. The search range for the optimal threshold in the bio-optimized region growing is from 0 to 1 with stepsize 0.01.

Normalized cut is a graph based approach. It considers each pixel as a vertex and edge as a connection weight between pixels. The main objective of this approach is to minimize the disassociation between the groups while maximizing the association within the groups. The number of possible groups is determined by the end user. As a result, the number of possible groups in an image is the adaptive parameter in the bio-optimized normalized cut. Since the bleb is the foreground and the rest of the image is the background, we have at least two components in the image. Based on observation, each frame consists of five different regions: 1) background; 2) cell body; 3) halo; 4) bleb; 5) debris or part of neighboring cell. Since the image contains these five basic regions, the number of expected regions is five under ideal condition. With the consideration of the worst case scenario, we double the number of possible components in an image. Therefore, the search range is set to be 2 to 10 components.

The meanshift method is a density based approach. It has two parameters: spatial and range bandwidths. It also requires the minimum size of a region. In this paper, the minimum size of a region is set to be 60 pixels which is the smallest recognizable bleb. The spatial range determines the size of the search window that computes the meanshift. The range bandwidth determines the window size that is used to compute the feature. In this paper, the optimization search range for the spatial bandwidth is from 1 to 8 and the range bandwidth is from 1 to 16.

Watershed is a topological based method. It is often applied on a gradient image. It partitions the image into two different sets: catchment basins and watershed line. The watershed method floods the topographic surface of a gradient image from its regional minima. It builds watershed lines to prevent waters in different catchment basins from merging. In this paper, the gradient image is the Euclidean distance transform of the marker image. The marker image (binary image) is obtained with the extended minimum of the original image approach [44]. The extended minimum approach depends on the suppressing value to binarize the image. Therefore, the suppressing value is the adaptive parameter for the bio-optimized watershed. Since the image is an 8 bit image, the search range for the optimal suppressing value is from 1 to 254.

*4.2.3.2 Bio-optimization metric*

The optimization metric $\Psi$ has two parameters; $\mathrm{Area}^{\mathrm{t}}(\mathrm{i})$ and $\mathrm{Area}_{\mathrm{Pre}}(\mathrm{t})$. The metric calculates the Euclidean distance between the areas obtained from the segmentation result at $\mathrm{i}_{\mathrm{th}}$ iteration and the predicted area at frame index $\mathrm{t}$.

$$\Psi^{\mathrm{t}}(\mathrm{i}) = \sqrt{\left(\mathrm{Area}^{\mathrm{t}}(\mathrm{i}) - \mathrm{Area}_{\mathrm{Pre}}(\mathrm{t})\right)^2} \tag{4.6}$$

The variable $\mathrm{Area}^{\mathrm{t}}(\mathrm{i})$ is the bleb area at the $\mathrm{i}_{\mathrm{th}}$ iteration under specific parameters. $\mathrm{Area}_{\mathrm{Pre}}(\mathrm{t})$ is the predicted area at the frame index $\mathrm{t}$ which is obtained from equation (4.7). Its value is assigned conditionally by the following equation.

$$\mathrm{Area}_{\mathrm{pre}}(\mathrm{t}) = \begin{cases} \mathrm{Area}_{\mathrm{init}} A_{\mathcal{P}_{\mathrm{e}}} \mathrm{t}^{B_{\mathcal{P}_{\mathrm{e}}}}, & \text{if expansion} \\ \mathrm{Area}_{\mathrm{init}} A_{\mathcal{P}_{\mathrm{r}}} \mathrm{t}^{B_{\mathcal{P}_{\mathrm{r}}}}, & \text{if retraction} \end{cases} \tag{4.7}$$

where $\mathrm{Area}_{\mathrm{init}}$ is an estimated initial intermediate bleb area. The final bio-optimized metric is shown below:

$$\Psi_{\mathrm{opt}}(t) = \min_i \Psi\left(\mathrm{Area}^t(i); \mathrm{Area}_{\mathrm{pre}}(t)\right) \quad (4.8)$$

The optimized $\Psi$ is found when at $i_{\mathrm{th}}$ iteration the metric value is minimized.

*4.2.3.3 Update Parameters and Output*

The initial seed point to locate the intermediate bleb is given by the end user. The subsequent seed points in a video sequence are generated automatically. The seed point of the next frame is predicted by finding the centroid of the detected bleb region at the current video frame [30]. The assumption of smooth/gradual transition between consecutive frames is made for both expansion and retraction processes. The detected bleb region is a region that has an area that minimized the bio-optimization metric with the predicted area at the current frame. The output of the method for a video is a sequence of binary masks of the detected blebs.

*4.2.3.4 Bleb Region Detection Algorithm Flowcharts*

The bleb region detection algorithm requires five inputs, and it will return a binary image sequence of segmented blebs. The five inputs are: 1) hESC bleb phase contrast video; 2) seed point to identify intermediate bleb initially; 3) intermediate bleb area; 4) intermediate bleb frame index; 5) model selection for segmentation. The interactive part occurs in finding either the seed point or the initial intermediate bleb area. These two parameters can be acquired in two ways: 1) user defines the seed point and obtains the intermediate bleb area through a conventional segmentation method; 2) user

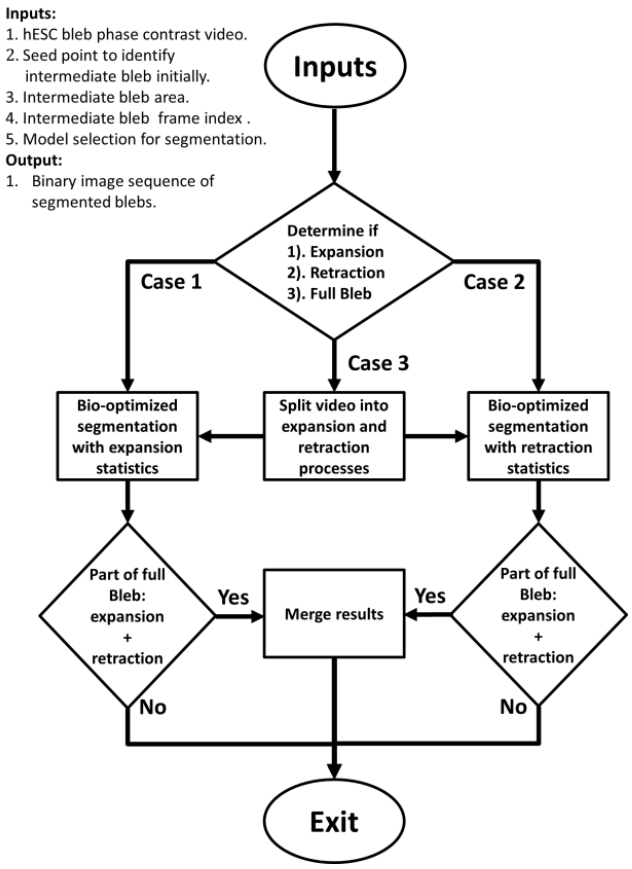



**Figure 17** Flowchart of the bleb region detection algorithm.

outlines the intermediate bleb region and uses the centroid of the bleb region as the seed point. In this paper, we chose the second approach for the seed point and intermediate bleb area. The first approach requires fine tuning of parameters in a conventional segmentation method to get a desirable segmentation result. Figure 17 shows the main flowchart of the algorithm. The bio-optimized segmentation method with retraction statistics is shown in Figure 18. The implementation for the bio-optimized segmentation with expansion statistics is similar to the flowchart shown in Figure 18 with two exceptions: 1) need to use expansion statistics; 2) expansion video sequence must be reversed to be similar as a retraction process.

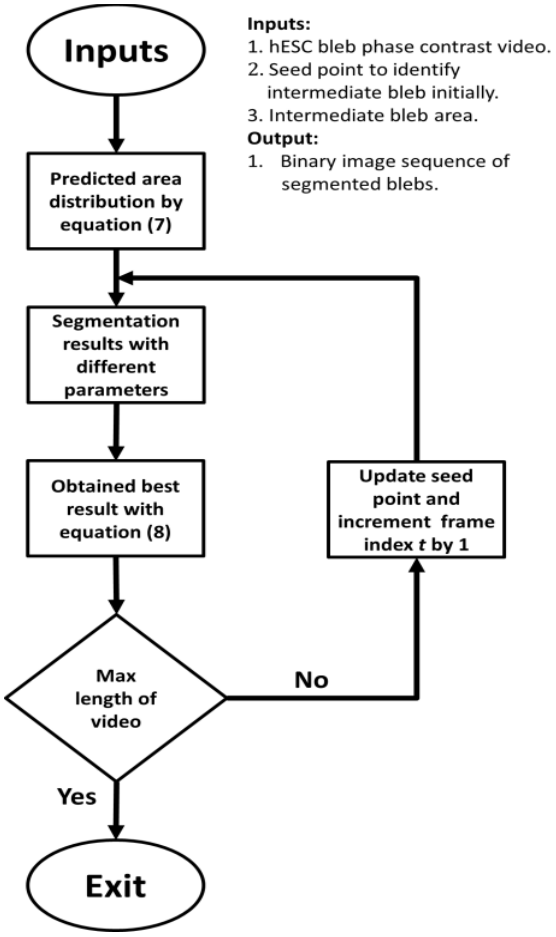



**Figure 18** Flowchart for the bio-optimized segmentation of the retraction process

*4.2.3.5 Model Selection for Segmentation*

The proposed method can be performed offline or online. The offline system will use the pre-determined intermediate bleb for its processes. For the online system, a correction for the wrongful detection of the intermediate bleb detection method is required. Since we have expansion and retraction processes, we have two basic cases and one full bleb formation case for the online system. If the bleb sequence is shrinking in size, then the system will use the bio-statistic parameters for the retraction process. However, if the bleb sequence is expanding, then the bio-statistic parameters for the

expansion process will be used. In the event of full bleb formation, the video will be split into expansion and retraction videos, which will be used in expansion and retraction processes, respectively. Since the video is split based on the intermediate bleb frame location, the intermediate bleb frame location must be accurate as it will affect the performance of both processes.

## 4.3. Experimental Results

### *4.3.1 Data*

All time lapse videos were obtained with a BioStation IM [32] [46]. The frames in the video are phase contrast images. The videos were acquired using 20x objectives with 600 x 800 resolutions. We analyze 60 cell videos of expansion and retraction which are cropped randomly from the original videos. There are 30 videos for expansion and 30 videos for retraction. Figure 19(a) shows the frame distribution for each video. The expansion videos have an average of 8.5 frames. The retraction videos have an average of 12.6 frames. The peak at Figure 19(a) for video #9 is a special phenomenon when the bleb neither expands nor retracts. Each video frame is taken 2 seconds apart. In addition, we also analyzed 29 sets of videos for bleb area distribution over time for expansion and retraction model parameters. Figure 19(b) shows the frame distribution of the training data set for the bio-statistic parameters, and it has an average of 17.3 frames. Further, nine new videos (not a part of the above dataset) of full bleb formation sequences were used to demonstrate the importance of the learned parameters for expansion and retraction. The ground-truth for all videos was generated manually by expert biologists.

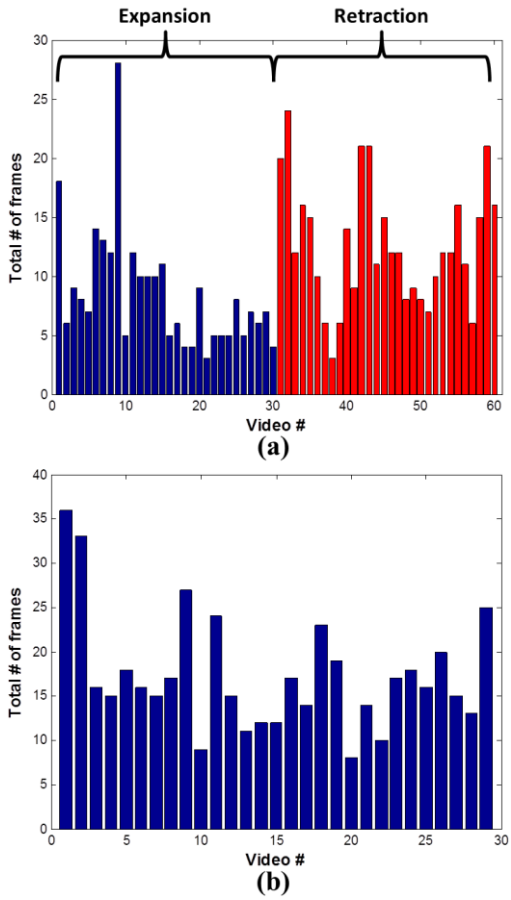



**Figure 19** (a) Frame distribution of experimental dataset; (b) frame distribution of the training dataset.

*4.3.2 Parameters*

The intermediate bleb area and seed point were needed initially for each video sequence and were provided by the end user. In the case of full bleb formation, intermediate frame number can either be given by the end user or predicted by 10/28 ratio. The estimated length of the expansion data is 10 frames and the estimated length of retraction training data is 18 frames. The estimated lengths are calculated by following equation.

$$T = \left\lceil \frac{T_{min} + T_{max}}{2} \right\rceil \quad (4.9)$$

where T is the final parameter search range index. $T_{min}$ and $T_{max}$ are the minimum and maximum number of frames of the expansion or retraction process, respectively, derived from the 29 sets of bleb formation sequences.

Since the power law function yields the lowest total mean squared error in model fitting, both expansion and retraction processes are characterized by a power law function with different parameters. For expansion, $A_{\mathcal{P}_e}$ and $B_{\mathcal{P}_e}$ have values 4.192 and 0.822, respectively. The parameters $A_{\mathcal{P}_r}$ and $B_{\mathcal{P}_r}$ of retraction process are 5.472 and 1.168.

*4.3.3 Performance Measures*

For comparison, we use methods suggested by Shattuck et al. [47]. Our comparison metrics are Jaccard similarity (JAC), Dice coefficient (DIC), sensitivity (SEN), and specificity (SPC). The JAC is a measure of similarity between experimental results and the ground-truth. The DIC is the measure of the agreement between experimental results and ground-truth. The SEN is a measurement of the proportion of actual positives which are correctly identified. The SPC measures the proportion of the actual negatives which are correctly identified. The equation for each metric is shown below:

$$\mathrm{JAC} = \frac{\mathrm{TP}}{(\mathrm{TP+FP+FN})} \tag{4.10}$$

$$\mathrm{DIC} = \frac{2\mathrm{TP}}{(2\mathrm{TP+FP+FN})} \tag{4.11}$$

$$\mathrm{SEN} = \frac{\mathrm{TP}}{(\mathrm{TP+FN})} \tag{4.12}$$

$$\mathrm{SPC} = \frac{\mathrm{TN}}{(\mathrm{FP+TN})} \tag{4.13}$$

The variables are defined as follows: 1) true positive (TP), 2) true negative (TN), 3) false positive (FP) and 4) false negative (FN). TP indicates the overlapped region of the detected bleb's binary mask and the bleb ground-truth's binary mask. TN is the overlapped region of the detected background's binary mask and the background ground-truth's binary mask. FP is the detected background's binary mask that is falsely identified as part of the bleb region. FN is the detected bleb's binary mask that is falsely identified as part of the background.

*4.3.4 Comparison of Methods for Expanding and Retracting Blebs*

The segmentation parameters are identified individually by a receiver operating characteristic (ROC) curve [31]. Figure 20 shows the ROC curves for four different segmentation methods for a single video of blebs. The optimal point is the maximum true positive rate within the range of 0 to 0.2 false positive rates. Based on the ROC, we determine that the optimal range and spatial bandwidths are 1 and 1 for meanshift. Its minimum region criterion is set to 60 pixels which is the smallest recognizable bleb size. For the normalized cut method, the optimal number of components is 5. For region growing, the optimal similarity threshold is set to be 0.05. For watershed, the optimal suppressing value is 180.

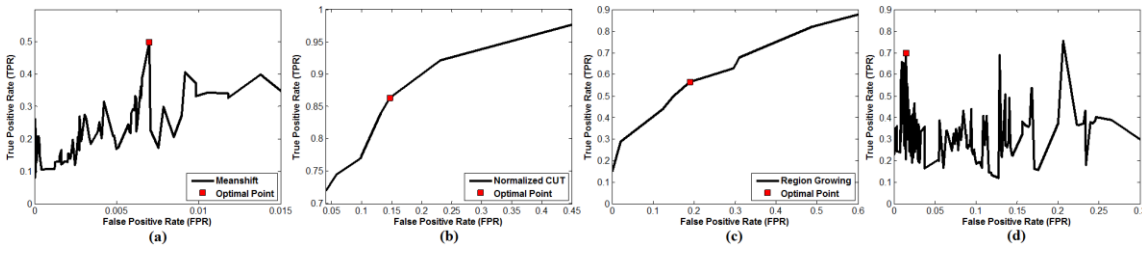



**Figure 20** ROC plot for (a) Meanshift, (b) normalized cut, (c) region growing, (d) watershed.

The proposed method was tested on a total of 60 expansion and retraction videos that were acquired with a 20x objective. The segmentation algorithms with the proposed method outperformed all conventional segmentation methods in this chapter. The bio-optimized meanshift outperformed the conventional meanshift by more than 24% in expansion and 12% in retraction for both JAC and DIC measures in mean performance as shown in Table 9. The meanshift with fixed parameters outperforms other conventional

**Table 9**
**Results on expansion and retraction videos (Note All Values are in Percentage)**

**Results on Expansion Videos**

| Metric | JAC | | | | DIC | | | | SEN | | | | SPC | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Method** | **Min** | **Max** | **Std** | **Mean** | **Min** | **Max** | **Std** | **Mean** | **Min** | **Max** | **Std** | **Mean** | **Min** | **Max** | **Std** | **Mean** |
| Meanshift* | **11.1** | **81.4** | 17.5 | 55.3 | **19.9** | **89.7** | 17.7 | 69.3 | 16.5 | 82.2 | 15.7 | 58.6 | 95.5 | **100.0** | 1.0 | 99.4 |
| Meanshift | 1.2 | 69.1 | 23.1 | 30.8 | 2.3 | 81.7 | 26.9 | 42.5 | 8.7 | **100.0** | 21.9 | 56.8 | 0.1 | **100.0** | 31.6 | 81.7 |
| NCUT* | 3.4 | 69.2 | 18.3 | 47.2 | 6.5 | 81.8 | 19.8 | 61.7 | 12.9 | 84.4 | 15.2 | 62.4 | 88.2 | **100.0** | 3.1 | 97.6 |
| NCUT | 1.1 | 67.5 | 20.8 | 27.9 | 2.3 | 80.6 | 25.0 | 39.7 | 7.4 | 89.8 | 18.6 | 59.2 | 37.9 | **100.0** | 13.6 | 87.2 |
| Region Growing* | 5.8 | 62.0 | 15.6 | 40.5 | 10.9 | 76.6 | 17.7 | 55.7 | 21.0 | 72.9 | **12.7** | 49.5 | 83.8 | **100.0** | 4.1 | 97.4 |
| Region Growing | 2.7 | 51.8 | 14.5 | 17.3 | 5.3 | 68.2 | 19.2 | 27.2 | 11.4 | **100.0** | 23.6 | 53.8 | 18.6 | **100.0** | 24.6 | 76.0 |
| Watershed* | 6.8 | 80.0 | 15.9 | **57.6** | 12.8 | 88.9 | 16.8 | **71.5** | 12.8 | 86.6 | 15.6 | 61.4 | **96.4** | **100.0** | **0.8** | **99.5** |
| Watershed | 1.5 | 32.0 | **8.2** | 9.9 | 2.9 | 48.5 | **12.6** | 17.2 | **46.4** | **100.0** | 17.7 | **82.5** | 0.0 | 87.5 | 33.1 | 39.0 |

**Results on Retraction Videos**

| Metric | JAC | | | | DIC | | | | SEN | | | | SPC | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Method** | **Min** | **Max** | **Std** | **Mean** | **Min** | **Max** | **Std** | **Mean** | **Min** | **Max** | **Std** | **Mean** | **Min** | **Max** | **Std** | **Mean** |
| Meanshift* | **12.2** | **70.5** | 14.0 | **52.5** | **21.8** | **82.7** | 14.5 | **67.6** | 16.3 | 71.1 | 12.7 | 54.5 | **96.1** | **100** | **0.8** | **99.6** |
| Meanshift | 5.0 | 69.5 | 17.6 | 40.3 | 9.5 | 82.0 | 19.1 | 55.2 | **35.2** | 74.0 | 9.9 | 54.9 | 61.2 | **100.0** | 8.4 | 95.5 |
| NCUT* | 1.4 | 62.3 | 17.9 | 39.6 | 2.7 | 76.8 | 21.0 | 54.2 | 2.7 | 81.0 | 19.6 | 53.9 | 92.7 | 99.9 | 2.1 | 97.4 |
| NCUT | 3.2 | 43.8 | 13.1 | 21.5 | 6.2 | 60.9 | 17.3 | 33.6 | 20.0 | 87.2 | 18.5 | 53.9 | 47.6 | 98.0 | 9.3 | 88.0 |
| Region Growing* | 3.7 | 60.5 | 14.2 | 35.9 | 7.1 | 75.4 | 16.4 | 51.3 | 29.7 | 68.4 | **9.2** | 46.8 | 79.5 | **100.0** | 4.7 | 97.2 |
| Region Growing | 2.1 | 30.1 | **7.1** | 13.2 | 4.0 | 46.2 | **11.0** | 22.7 | 15.2 | 99.9 | 25.1 | 54.9 | 21.8 | **100.0** | 23.2 | 72.5 |
| Watershed* | 7.2 | 67.6 | 16.3 | 46.7 | 13.5 | 80.7 | 18.3 | 61.7 | 9.6 | 70.8 | 15.2 | 48.8 | 94.9 | **100.0** | 1.4 | 99.3 |
| Watershed | 0.6 | 62.9 | 11.1 | 9.5 | 1.2 | 77.2 | 13.7 | 16.1 | 14.2 | **100.0** | 23.3 | **81.0** | 0.1 | 99.0 | 30.5 | 35.2 |

*** denotes a bio-optimization metric was used and bold numbers denote the best performance that was achieved.**

segmentation methods. The bio-optimized meanshift performed the best on average for both expansion and retraction results. All other bio-optimized methods outperformed their conventional counterparts by at least 18%. The bio-optimized meanshift and watershed can even have a maximum performance of above 80% in both JAC and DIC measures for the expansion case. Among all the compared methods, bio-optimized meanshift had the highest minimum and maximum performance in JAC and DIC measures.

Table 9 shows that the bio-optimized meanshift method has lower average standard deviation (Std) than the bio-optimized watershed which is the second best performer. Since the cell and the bleb are dynamically changing over time, adaptive parameters that are found by the bio-optimized metric improves the performance significantly. The fixed parameters in the conventional segmentation methods are not sufficient to handle the blebbing sequence where the bleb's image properties change over time.

A set of bleb images with results is provided in Figure 21. The proposed method yields a tight boundary for the bleb regions. However, the conventional segmentation methods with fixed parameters suffered more from severe over-segmentation. Due to the bio-optimized metric, the bio-optimized segmentation methods did not suffer from severe over-segmentation. The bio-optimized segmentation methods' solutions are bounded with the predicted area. Therefore, the bio-optimized segmentation methods have higher performance for the blebbing sequence than the conventional segmentation methods. The constant parameters in the conventional segmentation approaches are the cause for their

low performance. The proposed methods with adaptive parameters are able to capture the local region of the bleb more accurately.

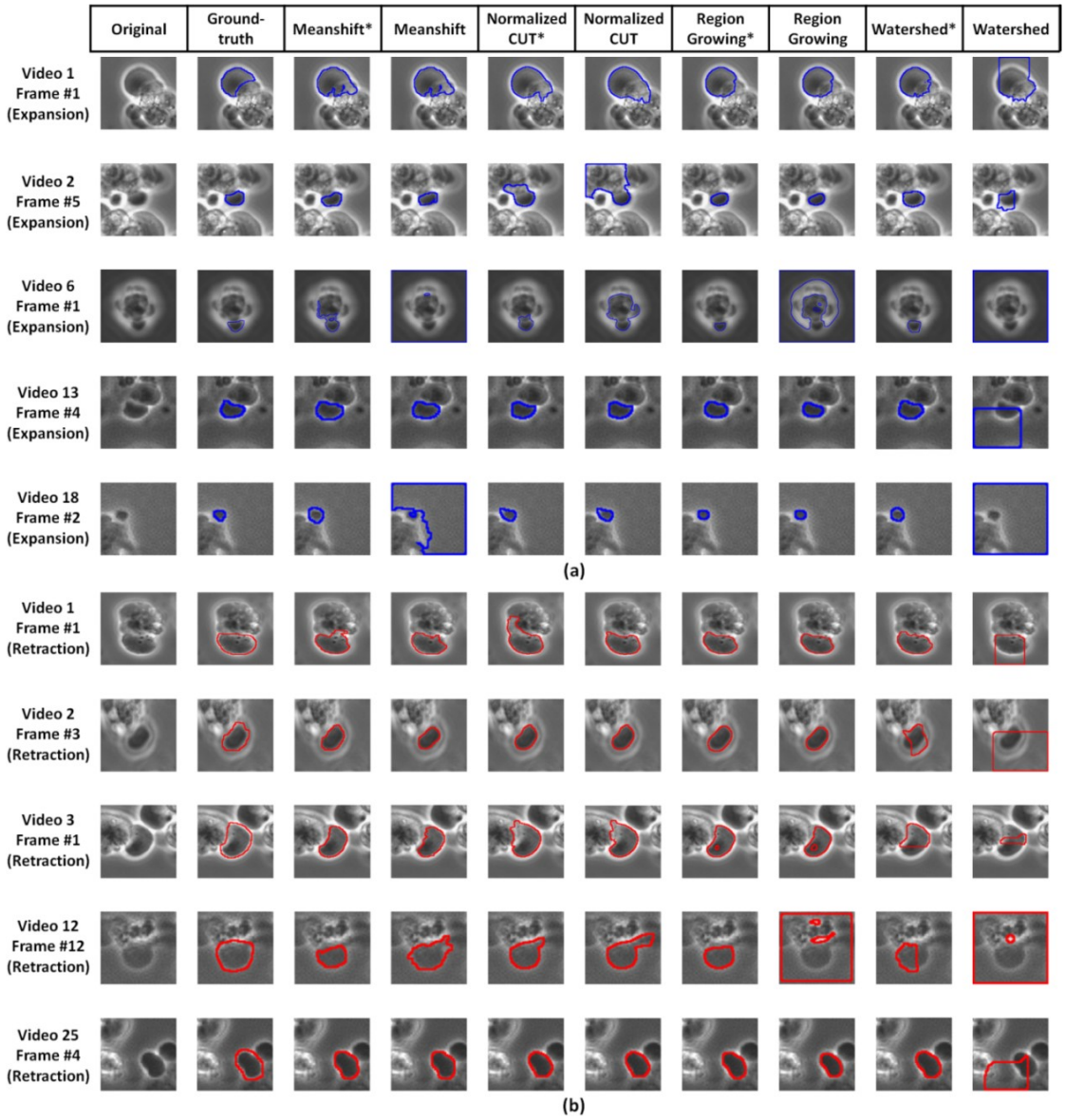



**Figure 21** Visual comparisons of ten different blebs with results from (a) expansion process and (b) retraction process.
(Note that frames from two sets of videos are used in the figure: expansion and retraction videos. Expansion results are outlined in blue and retraction results are outlined in red. )

#### *4.3.5 Proposed Method on Full Bleb Formation Sequence*

With the learned parameters for expansion and retraction, we tested the bio-optimized meanshift method on nine new videos with the full bleb formation sequences in which both expansion and retraction occurred. The location of the intermediate bleb frame was automatically estimated by the 10:18 frame ratio as mentioned in Section 4.3.2. Based on the aforementioned ratio, the intermediate bleb frame was estimated to be in 10/28 of the full bleb formation length of a video sequence. Once the frame that contains the intermediate bleb was determined, we started the expansion and retraction processes by locating the intermediate bleb with a single seed point (supplied by the user). We compared the above results with the results we obtained manually by outlining the intermediate bleb in a video frame. The results based on automated and manual intermediate bleb detections are shown in Table 10. The reasons for video #1 failed on both cases were: 1) no visible divisions between the bleb and cell body/neighboring blebs; 2) cell body was detected as part of the bleb. Despite the low performance in video #1, the overall performance is still above 50% for both approaches. The average measures by the two methods are quite close. The maximum average difference among all the measures is less than 2%.

### *4.3.6* Discussion

#### *4.3.6.1 Effect of Bio-statistic Parameters*

The bio-statistic parameters are essential in the optimization process. Inaccurate parameters might lead to either under-segmentation or over-segmentation of the bleb. As

**Table 10 Performances on automated and manual intermediate frame selection**

**Performance on Automated Frame Selection**

| Video # | JAC | DIC | SEN | SPC |
|---|---|---|---|---|
| Video 1 | 10.39% | 18.83% | 16.30% | 98.72% |
| Video 2 | 54.46% | 70.52% | 62.36% | 99.81% |
| Video 3 | 56.90% | 72.53% | 61.23% | 99.81% |
| Video 4 | 39.68% | 56.81% | 49.18% | 99.68% |
| Video 5 | 40.43% | 57.58% | 56.01% | 99.56% |
| Video 6 | 63.90% | 77.97% | 67.16% | 99.96% |
| Video 7 | 56.65% | 72.33% | 58.22% | 99.97% |
| Video 8 | 65.31% | 79.01% | 65.64% | 99.99% |
| Video 9 | 64.03% | 78.07% | 67.41% | 99.95% |

**Performance on Manual Frame Selection**

| Video # | JAC | DIC | SEN | SPC |
|---|---|---|---|---|
| Video 1 | 14.67% | 25.59% | 24.04% | 98.57% |
| Video 2 | 54.46% | 70.52% | 62.36% | 99.81% |
| Video 3 | 60.52% | 75.40% | 62.82% | 99.90% |
| Video 4 | 39.68% | 56.81% | 49.18% | 99.68% |
| Video 5 | 40.43% | 57.58% | 56.01% | 99.56% |
| Video 6 | 60.95% | 75.74% | 65.51% | 99.94% |
| Video 7 | 57.15% | 72.73% | 58.76% | 99.97% |
| Video 8 | 65.71% | 79.31% | 67.41% | 99.97% |
| Video 9 | 64.03% | 78.07% | 67.41% | 99.95% |

**Average Results on Nine Videos of Expansion and Retraction (Automated Frame Selection)**

| Metric | JAC | DIC | SEN | SPC |
|---|---|---|---|---|
| Mean | 50.19% | 64.85% | 55.95% | 99.72% |
| Max | 65.31% | 79.01% | 67.41% | 99.99% |
| Min | 10.39% | 18.83% | 16.30% | 98.72% |
| Std | 17.69% | 19.14% | 15.98% | 0.40% |

**Average Results on Nine Videos of Expansion and Retraction (Manual Frame Selection)**

| Metric | JAC | DIC | SEN | SPC |
|---|---|---|---|---|
| Mean | 50.84% | 65.75% | 57.05% | 99.71% |
| Max | 65.71% | 79.31% | 67.41% | 99.97% |
| Min | 14.67% | 25.59% | 24.04% | 98.57% |
| Std | 16.53% | 17.17% | 13.71% | 0.45% |

a result, the bio-statistic parameters found through model fitting are important for equation (4.8) which uses predicted area found by these bio-statistic parameters. The predicted area is used to evaluate the segmentation result at each iteration i for an optimal solution. There are no severe over-segmentations in the bio-optimized segmentation methods as shown in Figure 21. However, the conventional methods have at least one severe over-segmentation.

*4.3.6.2 Effect of the Bleb and Cell Sizes*

The bleb and cell body size are important parameters in the optimization process. There are three types of small blebs that have poor performance in the proposed method:

1) Small blebs due to smaller cell (typically a small bleb has roughly 240 pixels or less).
2) Small blebs at the initial stage of the expansion.
3) Small blebs at the final stage of the retraction.

The smaller blebs for any process are not fully developed and often have similar intensity and texture as the cell body. Therefore, over-segmentation is bound to happen if the cell body and its bleb share similar image properties. The proposed method yields a lower value in comparison metric for small blebs. Larger cells with larger blebs perform better. A typical large bleb has 1700 or more pixels.

*4.3.6.3 Automated Segmentation*

The bio-optimized segmentation method is an automated approach in which the seed is given by the user initially. Since computationally expansion and retraction processes are modeled as inverse processes with different parameters for the bio-optimized segmentation method, the following facts must be true for the cases shown in Figure 14. For expansion, video sequence reversal must be done to make sure it is computationally the same as retraction process. For the full bleb formation process, video must be split into expansion and retraction videos. The separation is done by following the 10/28 frame ratio for the estimation of intermediate bleb frame location. This interactive system is a vital resource for biologists to analyze bleb behavior. It alleviates the biologist's burden in complete manual extraction of bleb area distribution and provides the bleb boundary over time. It is a useful data mining approach to help biologist quantify analyses on dynamic and apoptotic blebbing behavior.

*4.3.6.4 Time Complexity*

Since the proposed method was an iterative optimization process, it yielded higher time complexity for all bio-optimized segmentation methods. The best performer among

the bio-optimized segmentation methods was the bio-optimized meanshift. However, it required an average of 11 seconds to process a single frame as shown in Table 11. The bio-optimized watershed which was the second best performer had the lowest time complexity in all bio-optimized methods. The tradeoffs between the bio-optimized meanshift and bio-optimized watershed were performance and time complexity. The experiments were done on a desktop with an Intel(R) Core™ i7-3770 CPU processor that run at 3.40 GHz.

**Table 11**
**Computational Time Statistics Per frame in seconds**

| Method | Min | Max | Mean | Std |
|---|---|---|---|---|
| Meanshift* | 2.073 | 94.345 | 11.036 | 17.888 |
| Meanshift | 0.006 | 0.090 | 0.020 | 0.020 |
| NCUT* | 0.883 | 103.818 | 9.864 | 20.908 |
| NCUT | 0.097 | 17.143 | 1.256 | 3.007 |
| Region Growing* | 7.539 | 399.599 | 47.350 | 82.738 |
| Region Growing | 0.007 | 5.506 | 0.539 | 1.132 |
| Watershed* | 0.658 | 13.303 | 2.162 | 2.712 |
| Watershed | 0.004 | 0.056 | 0.012 | 0.012 |

### 4.4. Conclusion

The proposed method uses parameters learned from bio-statistics to segment blebs in videos. The bio-optimized metric depends on the predicted area of the bleb and the parameters for the area prediction are learned by fitting the bleb area curve over time. The proposed method segments out the bleb with adaptive parameters rather with fixed parameters for all frames in a video. With the predicted area as a constraint, there are few cases of over-segmentation in the bio-optimized segmentation methods. Since the bio-

optimized approach is an attempt to find a solution that optimized the bio-optimization metric, it depends on the results from segmentation method under different parameters. However, the segmentation method might not generate an ideal/exact solution to optimize the metric tightly. The over-segmentation in the approach is the inability of the segmentation method to give a better solution. In terms of biological contribution, this paper introduces a new concept that the bleb formation/retraction process can be used as a biological indicator of cell health. Healthy cells retract their blebs back to the cell body, while non-healthy cells do not retract them or retract them slowly. In terms of a computational contribution, this paper suggests a bio-optimization metric to segment bleb regions. It introduces an approach to improve bleb detection accuracy by using adaptive parameters instead of using constant parameters for all bleb frames in a video. The proposed segmentation methods with adaptive parameters found by the bio-optimized metric has a higher accuracy as well as a consistently higher performance.

This work can be used by biologists to evaluate the state of health of hESC in culture in various experimental conditions. It could be valuable in drug screening and in toxicological studies where short times to an endpoint are desirable, as well as resource and time saving. It may be adaptable in the future to high throughput screening of chemicals and drugs that need to be evaluated for embryotoxicity. In the future, we will explore additional features for automated intermediate bleb detection.

# CHAPTER 5 hESC Classification: Random Network with Autoencoded Feature Extractor

Human embryonic stem cells (hESC) are derived from the inner cell mass of developing blastocysts and possess two important properties: 1) self-renewal and 2) pluripotency [21] [22] [48]. Self-renewal is the ability to go through unlimited cycles of cell division, and pluripotency is the capability to differentiate into any cell types in the human body. hESC are an important resource for regenerative medicine, basic research on human prenatal development, and toxicological testing of drugs and environmental chemicals. Under their state of pluripotency, they can also be maintained indefinitely [24] [35]. Therefore, understanding the behavior of hESC is fundamental for medicinal and toxicological research [24] [25] [46].

The classification of hESC in video is essential for quantifiable analysis of hESC processes and behavior [15]. However, manual analysis of stem cells is laborious and tedious, and is often inaccurate due to the three human limitations. First, the accuracy of a human performing classification is inversely proportional to long working hours. Second, the uncertainty in classification occurs due to a wide variety of appearances of objects in a class. Third, the amount of time put off from working on datasets can lead to confusion in discerning hESC into the right classes. Fig. 22 provides a modularized system overview for an automated segmentation and classification process. In this chapter, we focus essentially on the classification of the detected components from hESC videos; the detected components are the six general classes shown in Fig. 22. Guan et al. [48]

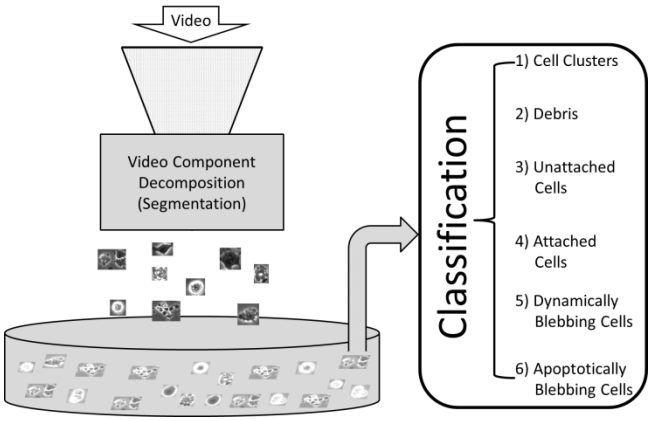



**Figure 22** System Overview.

provide details of a method for the fast detection and segmentation of individual video components.

Because phase contrast imaging is a non-invasive microscopy technique, it is widely used to study the behavior of live hESC in videos [49]. In this study, the hESC videos were taken with a BioStation IM [32]. The hESC videos consist of frames of phase contrast images. Each frame can contain any of the following six general components: 1) cell clusters, 2) debris, 3) unattached cells, 4) attached cells, 5) dynamically blebbing cells, and 6) apoptotically blebbing cells. Among these unattached, attached, dynamically blebbing and apoptotically blebbing cells are the four classes that are of significant interest in experimental work. These four classes are regarded as the four intrinsic cell types in a video. Fig. 23 shows examples of the six classes. Conceptually, the six classes of hESC can be distinguished with three fundamental human perceptual capabilities for identification and classification of objects: 1) shape, 2) intensity, and 3) texture. Each class can be uniquely identified by one or a combination of the aforementioned human perceptions. For instance, the apoptotically blebbing cells in

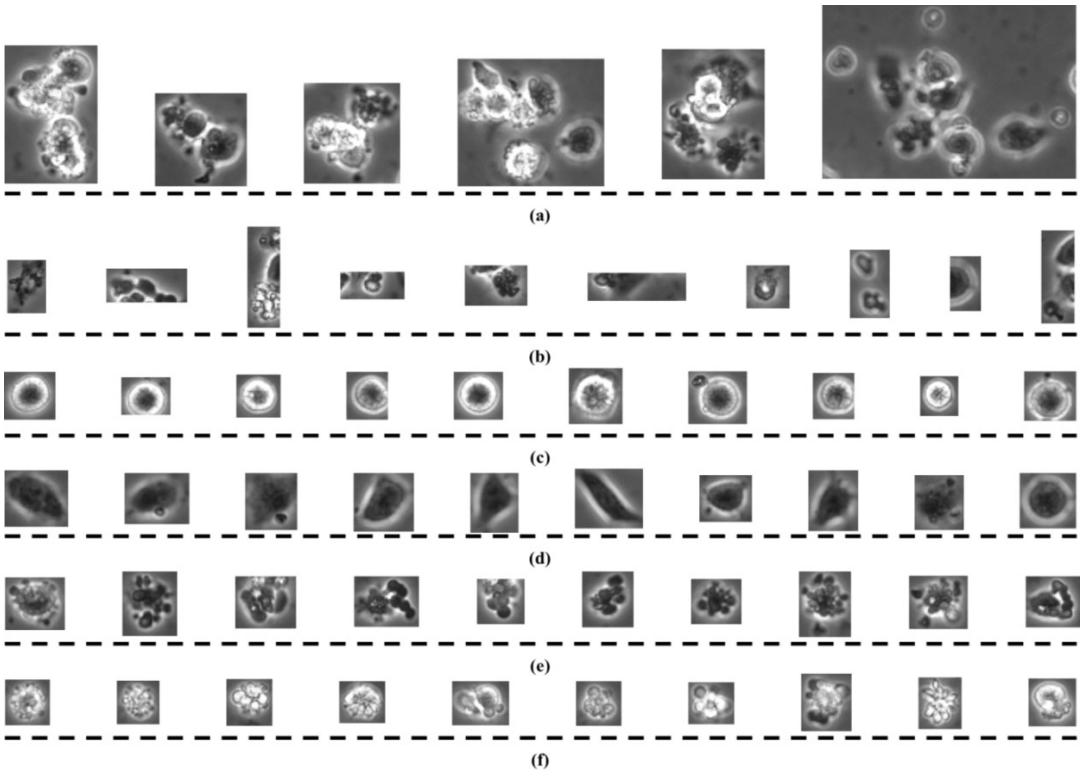



**Figure 23** Six general classes in a phase contrast frame: (a) Cell clusters, (b) debris, (c) unattached cells, (d) attached cells, (e) dynamically blebbing cells, (f) apoptotically blebbing cells.

Fig. 23(f) are similar in intensity, shape and texture among themselves. hESC in Fig. 23(e) and Fig. 23(f) are dissimilar in intensity, but they are similar in shape and texture. The debris in Fig. 23(b) has similar intensity values as various classes shown in Fig. 23. Traditionally, a feature vector can be derived with the aforementioned human perceptions. However, with the advent of deep learning techniques, we can develop classification models with the given abundancy of labeled data. Therefore, the need to generate a feature vector manually for classification system is only suitable when data is limited.

With the consideration that we often see abundant of unlabeled data rather than labeled data, we propose a random network with autoencoded feature extractor. The

proposed method focuses on building random subnetworks with feature extractor derived with unlabeled data. Moreover, the proposed method incorporated ensemble methodology in the network to reduce overfitting.

Section 5.1 discusses the motivation of this chapter. Section 5.2 presents the technical approach in detail. Section 5.3 provides experimental results and a discussion of the proposed and compared methods. Finally, Section 5.4 presents the conclusions of the chapter.

## 5.1 Motivation

In this Chapter, we first present the motivation for our proposed approach. This is followed by a method for automated cell region detection which is the segmentation component. We then describe random network (RandNet), and elaborate on the autoencoded feature extractor as well as the pre-trained subnetworks for the classification component. The classification component is part of the modularized system as shown in Fig. 24. A pseudo code for building the RandNet model is also provided.

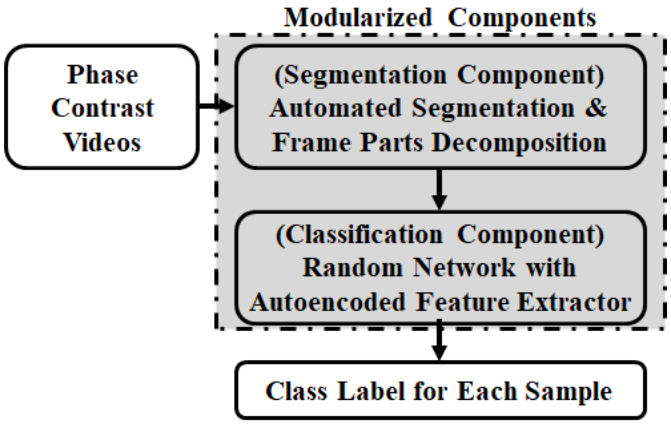



**Figure 24** Automated Segmentation & Classification System Overview.

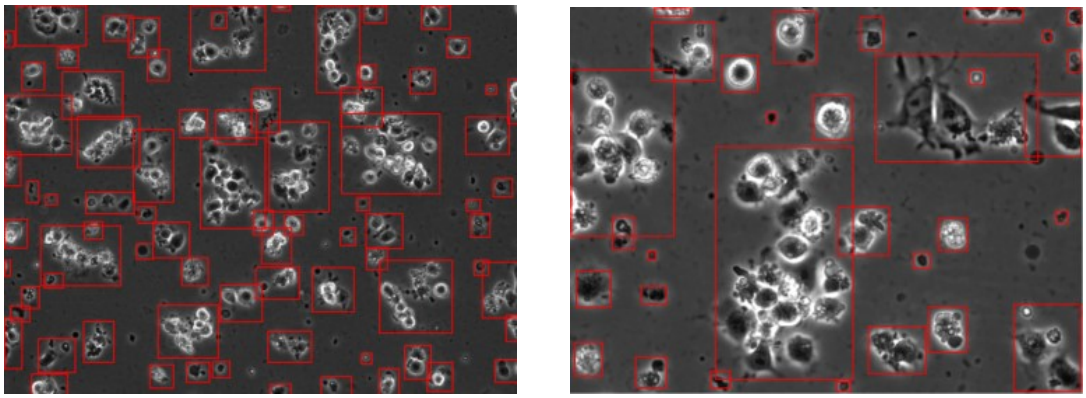
Domain knowledge often comes from human perception, which is the most complex yet efficient cognitive system. Through hypothetical assumption and visual inspection, we can sometimes identify useful features of hESC for classification. However, our domain knowledge is limited by the amount of information our brain can absorb. With tens of thousands of unlabeled and labeled data, experts can have hard times in either conceptualizing or generalizing the hidden information given by the data. Deep learning techniques can help us in understanding the vast amount of data and solve the difficulty in creating algorithms for human level repetitious tasks. The proposed approach uses an unsupervised technique to build the foundation of the encoder network. The proposed method also makes use of both unlabeled and labeled data to build a reliable classification system.

## 5.2 Technical Approach

### *5.2.1 Segmentation Component*

Guan et al. [48] proposed a model based method for automatically segmenting hESC. This automated cell region detection is an essential algorithm in developing automated frame component decomposition in hESC phase contrast videos. They considered the foreground and background intensity distribution as a mixture of two Gaussians. The objective of their algorithm is to find an optimal threshold that optimizes a criterion derived from the intensity distribution of foreground and background. The optimal segmentation is achieved at the highest criterion value. Since the segmentation method yields a binary image for each frame, we were able to extract a pool of individual

components from each frame. Fig. 25 shows the detected components of frames under 10x and 20x objectives. These detected components are then ready to be classified into one of the six aforementioned classes.



(a) (b)

**Figure 25** Detected Components on each Frame by Reference [48] (a) image under 10x objective, (b) image under 20x objective.

*5.2.2 Classification Component*

*5.2.2.1 Classification System Overview*

The proposed classification system was built with both labeled and unlabeled data, and it consists of many random pre-trained subnetworks. The proposed method utilizes unlabeled data to build the encoder component in the pre-trained subnetworks and labeled data to fine-tune the RandNet. RandNet structure also incorporated ensemble methodology to constrain overfitting. Figure 26 shows a graphical depiction of how RandNet is built with pre-trained subnetworks and the ensemble concept.

*5.2.2.2 Random Network*

Random network utilizes the concept of bagging in deep learning by creating subnetworks. Bagging or Bootstrap aggregation is a machine learning concept used to

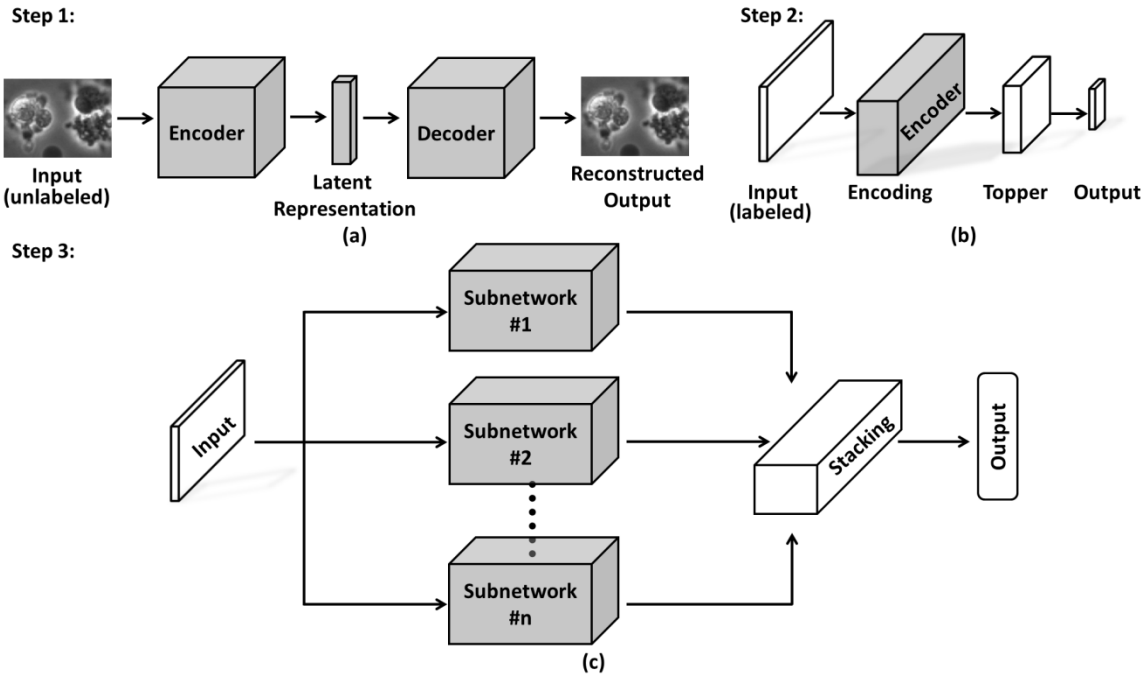



**Figure 26** (a) Autoencoder network, (b) pre-trained subnetwork, (c) random network (RandNet) with autoencoded feature extractor.

reduce variance and to avoid overfitting [50] [51] [52] [53]. RandNet, developed in this paper, is a method that contains many subnetworks that have a common pre-trained model and are fine-tuned with random samples. RandNet uses all the results from each subnetwork and passes it to a stacking network in which it makes the final decision. The detail of the stacking network is shown in Figure 27. The stacking network is designed to be simple and has only two main dense layers.

*5.2.2.3 Autoencoded Feature Extractor*

Autoencoder network is an efficient unsupervised learning method that learns the representation of a set of data. Autoencoder network contains two major components: encoder and decoder [54] [55] [56]. In this chapter, we used a structure similar to AlexNet as a basis of an encoder, and then we designed a decoder network from it;

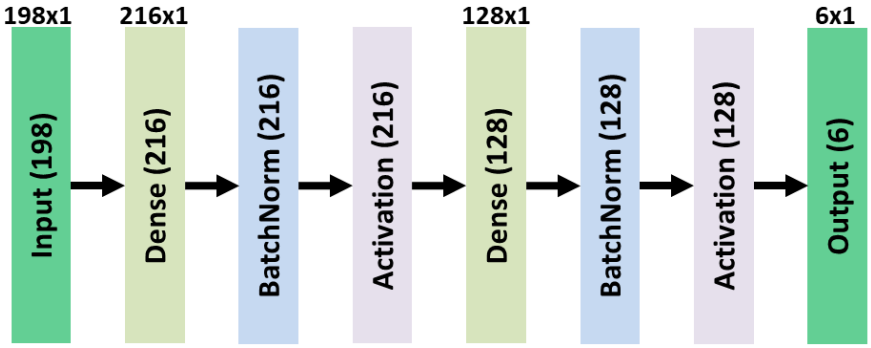



**Figure 27** Stacking Network

AlexNet structure was chosen for its simple implementation [57]. As shown in Figure 26(a), the encoder generates a set of latent representation for the unlabeled data. The details of both encoder and decoder structures are shown in Fig. 28. The autoencoder network used Adadelta optimizer and the pixel-wise binary cross-entropy loss function. Since the final layer in the autoencoder network was chosen to be a sigmoid activation layer, pixel-wise binary cross-entropy is an applicable loss measure. The loss function equation is given as the following:

$$\mathrm{Loss_{AE}} = -\sum_{r=1}^{N_R}\sum_{c=1}^{N_C} I(r,c)\log\big(K(r,c)\big) + \big(1 - I(r,c)\big)\log\big(1 - K(r,c)\big) \quad (5.1)$$

where $\mathrm{Loss_{AE}}$ is the total pixel-wise loss in autoencoder network, $N_R$ and $N_C$ are the total number of rows and columns. $I(r,c)$ and $K(r,c)$ are the truth and predicted label values in the $r_{th}$ row and $c_{th}$ column. Both $I(r,c)$ and $K(r,c) \in [0,1]$.

*5.2.2.4 Pre-trained Subnetwork*

The subnetwork used the encoder structure derived from the autoencoder network (in Step 2, Figure 26(b)) as the basis for building a subclassifier. Each pre-trained subnetwork is fine-tuned with random samples and has a topper structure. The layers of the topper structure are shown in Figure 29. Since the encoder structure was unfrozen in

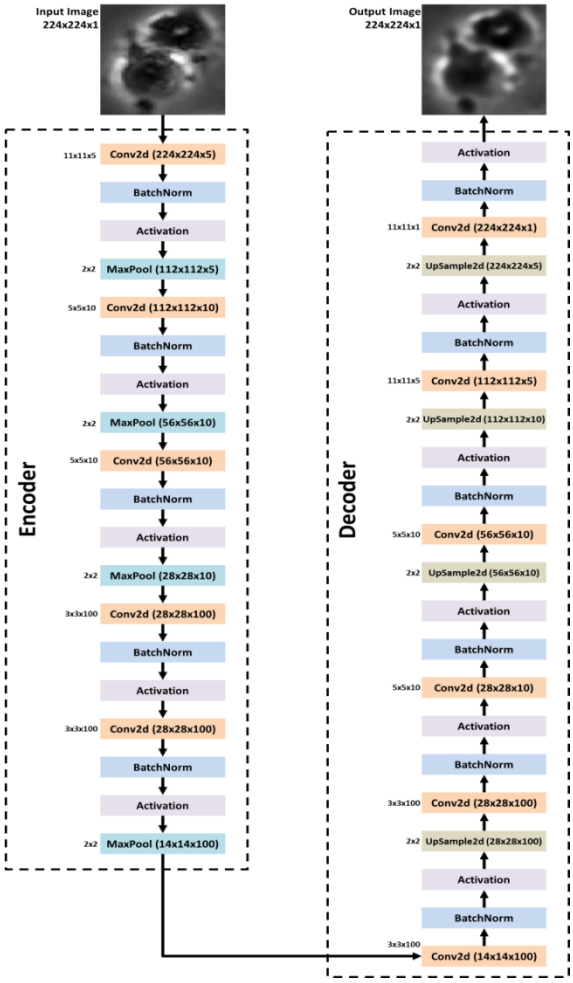



**Figure 28** Autoencoder network

each subnetwork, the fine-tuning with random samples affects the weights in the encoder structure. Therefore, we were able to emulate bagging for the proposed method. For this subnetwork, we use categorical cross entropy as our loss function which is shown below.

$$\text{Loss}_{\text{CCE}} = -\sum_{i=1}^{N_S}\sum_{j=1}^{M} y_{i,j}\log\left(p_{i,j}\right) \qquad (5.2)$$

where $\text{Loss}_{\text{CCE}}$ is the total categorical cross entropy in the pre-trained subnetwork. $N_S$ and M are the total number of samples and classes, respectively. $y_{i,j}$ and $p_{i,j}$ are the truth and predicted values for $i_{th}$ sample and $j_{th}$ class. The pseudo algorithm is given in Algorithm 2.

**Algorithm 2: Random Network with Autoencoded Feature Extractor**

Input: $\boldsymbol{D_{unlabeled}}$ is a set of unlabeled data
$\boldsymbol{D_{labeled}}$ is a set of labeled data
$\boldsymbol{n}$ is total number of subnetworks
Output: Final learned model.
1: **procedure** RandNet($\boldsymbol{D_{unlabeled}}$, $\boldsymbol{D_{labeled}}$, $\boldsymbol{n}$)
2: Normalized $\boldsymbol{D_{unlabeled}}$ and $\boldsymbol{D_{labeled}}$ by dividing by 255.
3: Train autoencoder network with unlabeled data (Step 1, Figure 26a).
4: Extract encoder structures from autoenocder network for subnetwork training (Step 2, Figure 26b).
5: Create $\boldsymbol{n}$ subset of labeled data with stratified bootstrap [58]. These subsets are used to obtain $\boldsymbol{n}$ subnetworks.
6: Fine-tune $\boldsymbol{n}$ subnetworks with the above datasets.
7: Connect the outputs from $\boldsymbol{n}$ subnetworks with stacking technique (Step 3, Figure 26c) [59].
8: Train the final network with all the training data.
9: Obtain the final model.
10: **end procedure**

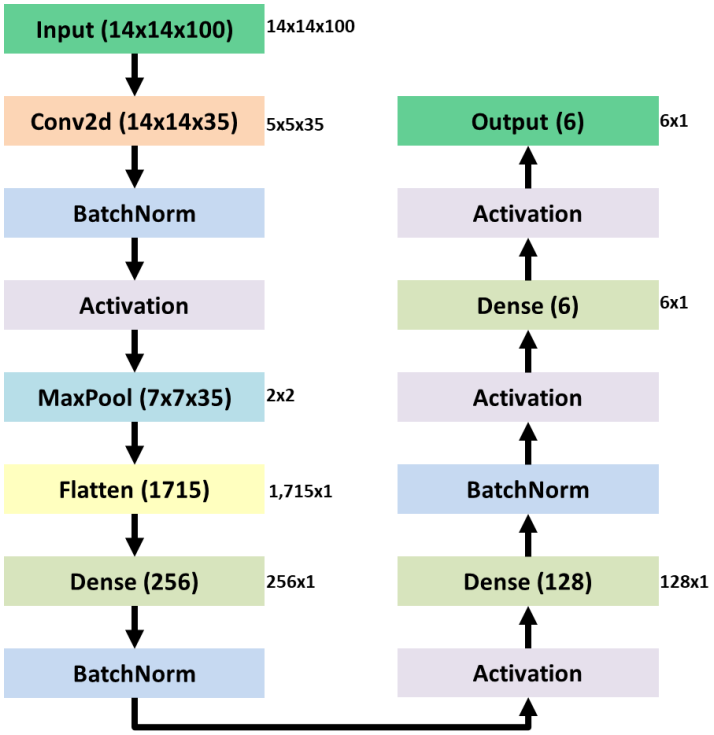



**Figure 29** Topper structure

## 5.3 Experimental Results

### *5.3.1 Data*

All time lapse videos were obtained with the phase contrast microscope in BioStation IM [32] [46]. The videos were acquired using either 10x or 20x objective with 600 x 800 pixel resolution. A total of 27,603 unlabeled gray scale images and 3,559 labeled gray scale images were obtained from six 10x videos and eight 20x videos. Both unlabeled and labeled images were obtained automatically by the method described in Guan et al. [10] [36] [48]. The labeled dataset had the following numbers of gray scale images for each class: 1) 636 cell cluster images; 2) 773 debris images; 3) 519 unattached cell images; 4) 704 attached cell images; 5) 413 dynamically blebbing cell images; and 6) 514 apoptotically blebbing cell images. The ground-truth for the datasets were generated manually by stem cell experts. We used 75% of the dataset for training and the remaining 25% of the dataset for out of sample testing for each class. In order to generalize the classifier, five-fold cross validation was done during model learning. Model learning is performed with training data only. The detail of 5-Fold cross validation is given in Figure 30.

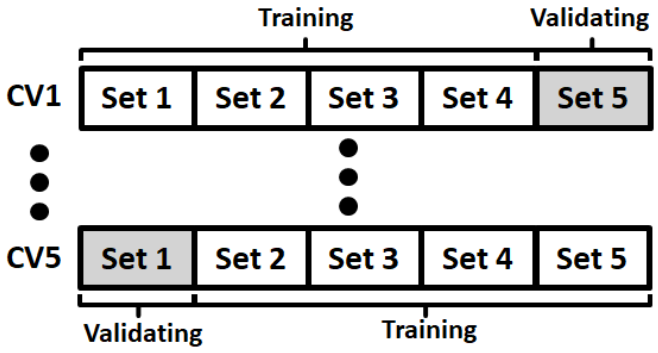



**Figure 30** 5-Fold cross validation scheme. (Note: the training data are splitted into 5 sets for the purpose of model selection).

*5.3.2 Parameters and Optimization*

For autoencoder network, there are two fixed parameters; epochs and batch size. The epochs and batch size are set to be 10 and 128 respectively. The default adadelta optimizer is used for the autoencoder network [60]. For RandNet, there are five parameters; epochs, batch size, number of subnetworks, learning rate and decay rate. We used 25 epochs with early stopping, a batch size of 50 and a total of 33 subnetworks. We also used a default adam optimizer with the learning rate of 0.001 and decay rate of 0 [61]. All parameters are fixed except the number of subnetworks. The number of the subnetworks has a search range from 1 to 37 with the step size of 2. Figure 31 shows that when the number of subnetworks equals to 33, it has the highest average validation accuracy as well as the lowest average validation loss.

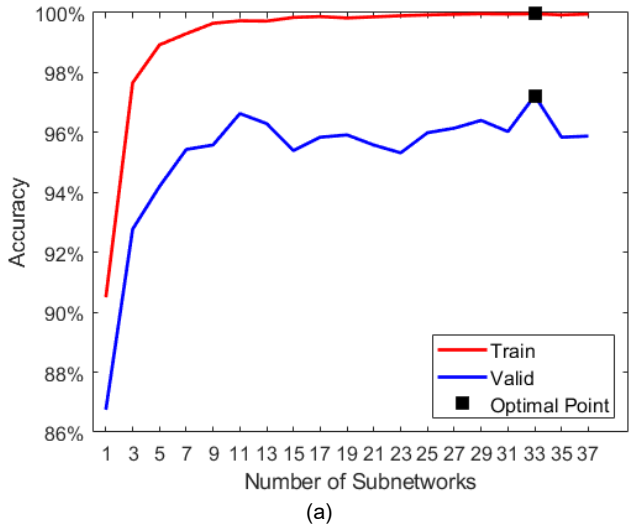
*5.3.3 Performance Measures*

For performance analysis and comparison, we used the confusion matrix for evaluation [62]. The following equations show the calculations for the overall and individual classification accuracy from the confusion matrix. Average classification rate (ACR) and individual true positive rate (TPR) are given by the following equations:

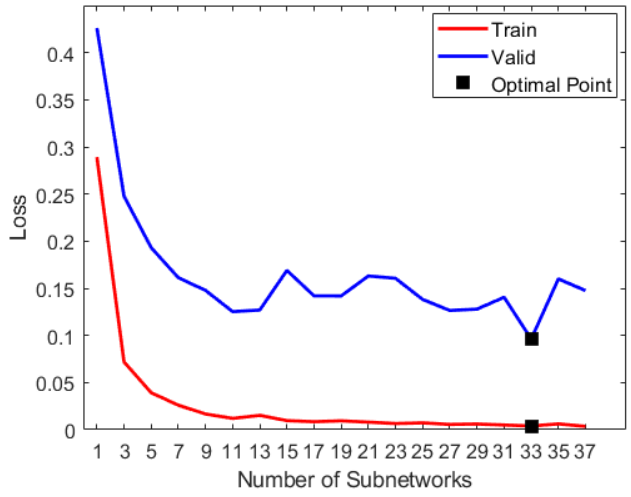
$$\mathrm{ACR} = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}_{\mathrm{class}}} \mathrm{CM}_{\mathrm{ii}} \tag{5.3}$$

$$\mathrm{TPR}_{\mathrm{j}} = \frac{1}{\mathrm{N}_{\mathrm{j}}} \sum_{\mathrm{i}=1}^{\mathrm{N}_{\mathrm{class}}} \mathrm{CM}_{\mathrm{ij}} \tag{5.4}$$

Note that $\mathrm{CM}_{\mathrm{ii}}$ is an $\mathrm{ii}_{\mathrm{th}}$ element in the confusion matrix $\mathrm{CM}$. $\mathrm{CM}$ is an element of $\mathbb{R}^{\mathrm{N}_{\mathrm{class}} \times \mathrm{N}_{\mathrm{class}}}$ where $\mathrm{N}_{\mathrm{class}}$ is the total number of classes. $\mathrm{N}$ is the total number of evaluated observations. $\mathrm{TPR}_{\mathrm{j}}$ is the true positive rate/ recall for the $\mathrm{j}_{\mathrm{th}}$ class. $\mathrm{N}_{\mathrm{j}}$ is the




(a)




(b)

**Figure 31** Mean accuracy vs. number of subnetworks curve; (b) mean loss vs. number of subnetworks curve. (5-Fold cross validation results)

total number of samples in the $j_{th}$ class. $CM_{ij}$ is the element of CM in the $i_{th}$ row and $j_{th}$ column.

There are three different categories of accuracies in evaluating the performance of a model, and they are: 1) training accuracies; 2) validation accuracies; and 3) out-sample testing accuracy. Training and validation accuracies refer to cross validation accuracies for training and validating sets, respectively. For example, in a 5-fold cross validation

scheme, the model is trained with the training set (see Figure 30). The out-sample testing accuracy is slightly different than the validation scheme. Once the best model parameters are learned from the model selection process, the final model is obtained with all the training dataset and the best parameters. This final model is then used to evaluate the performances of the testing dataset, and it produces the out-sample accuracy. Typically, training and validation accuracies show us the estimated bias and variance in the final model while out-sample testing accuracy shows the true variance in the final model.

### *5.3.4 Experimental Results and Discussions*

The proposed RandNet was compared with state-of-the-art methods as reported in Table 12. The top two performers are the proposed RandNet and the fused CNN triplet [20]. The proposed RandNet has 97.23% mean accuracy in a 5-fold cross validation and a seemingly low standard deviation in its validation results. The reason that both RandNet and fused CNN triplet outperformed other methods is that additional data are being used. Both aforementioned methods trained with data other than the given labeled data. The RandNet used unlabeled data to pre-train its subnetworks and then fine-tuned it with the

**Table 12 5-fold Cross validation results**

| Method | Mean Accuracy | STD |
|---|---|---|
| Fused CNN Triplet [20] | 95.14% | 0.91% |
| ResNet18 [63] | 92.16% | 2.25% |
| ResNet34 [63] | 92.52% | 3.34% |
| ResNet50 [63] | 89.38% | 2.26% |
| VGG-16 [64] | 91.78% | 3.35% |
| VGG-19 [64] | 93.60% | 2.48% |
| AlexNet [57] | 93.23% | 2.98% |
| RandNet | **97.23%** | **0.94%** |

**Table 13 Testing Data results**

| Method | Accuracy |
|---|---|
| Fused CNN Triplet [20] | 95.83% |
| ResNet18 [63] | 87.59% |
| ResNet34 [63] | 88.20% |
| ResNet50 [63] | 86.17% |
| VGG-16 [64] | 88.29% |
| VGG-19 [64] | 89.46% |
| AlexNet [57] | 87.41% |
| RandNet | **96.28%** |

labeled data. On the other hand, fused CNN triplet uses both synthetic data and real labeled data in training. ResNets, VGGs and AlexNet were trained with only labeled data. Consequently, they seem to have higher variance in their performance. They also deem to perform worst in out-sample testing which is shown in Table 13.

When comparing to ResNets, VGGs and AlexNet, the proposed method outperformed those methods by at least 6% as shown in Table 13. Their performances of these other methods were all closely within their individual standard. The proposed method has significant lower standard deviation than ResNets, VGGs and AlexNet. Therefore, the proposed method still performed better in out-sample testing. Since the proposed method incorporated the concept of bagging and used 33 random subnetworks, the proposed method was going to have a low standard deviation in performance.

When comparing to fused CNN triplet, RandNet outperformed fused CNN triplet in both 5-fold cross validation and out-sample testing. As shown in Table 12, RandNet was about 2% better than fused CNN triplet in validation results. In term of out-sample testing, the proposed method had a slight 0.45% lead on fused CNN triplet as shown in Table 13. The confusion matrix of the proposed method on testing dataset is shown in

**Table 14 Confusion Matrix for Testing Data with RandNet (Proposed Approach)**

| Predicted \ Truth | Cell Cluster | Debris | Unattached Cell | Attached Cell | Dynamically Blebbing Cell | Apoptotically Blebbing Cell |
|---|---|---|---|---|---|---|
| Cell Cluster | 154 | 0 | 0 | 2 | 3 | 1 |
| Debris | 0 | 187 | 1 | 0 | 0 | 0 |
| Unattached Cell | 0 | 0 | 121 | 0 | 0 | 1 |
| Attached Cell | 0 | 0 | 6 | 173 | 2 | 0 |
| Dynamically Blebbing Cell | 1 | 5 | 0 | 1 | 97 | 3 |
| Apoptotically Blebbing Cell | 4 | 1 | 1 | 0 | 1 | 123 |

**Table 15 Individual class recall for RandNet**

| Method | Recall |
|---|---|
| Cell Cluster | 96.86% |
| Debris | 96.89% |
| Unattached Cell | 93.80% |
| Attached Cell | 98.30% |
| Dynamically Blebbing Cell | 94.17% |
| Apoptotically Blebbing Cell | 96.09% |

Table 14. The proposed method also outperformed fused CNN triplet in term of training cost. RandNet's computational cost in training is significantly lower than the fused CNN triplet. According to Theagarajan et al. [20], fused CNN triplet used additional 240,000 synthetic images for training; 40,000 for each class. Fused CNN triplet took about a month for synthetic image generation and about four days for final model building. On the other hand, the proposed RandNet had about 5 hours of training time, and used only 27,603 unlabeled images for pre-training the decoder network. The proposed method was implemented on a desktop with 3.4GHz Intel(R) Core i7-3770 CPU and NVIDIA GeForce GTX 1070 GPU.

*5.3.4.1 Misclassification Examples*

The proposed method had at least 93% TPR/recall for each individual class as shown in Table 15. It performed better in identifying attached cells; it has a total of

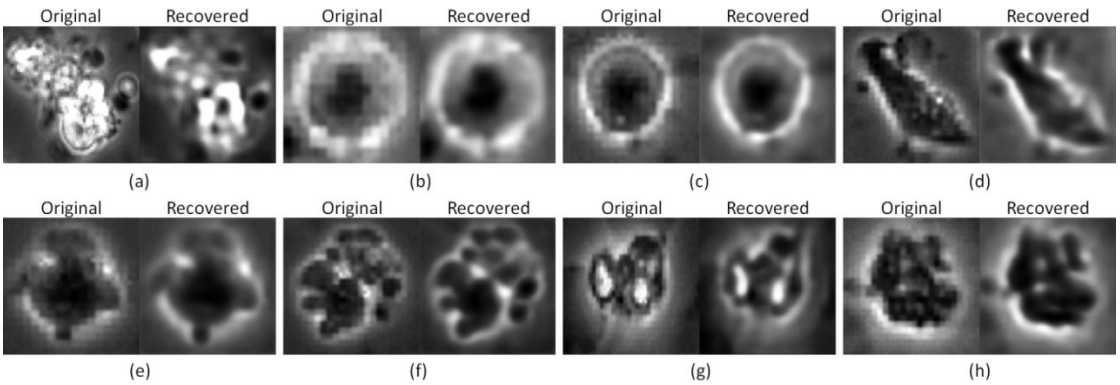



**Figure 32** Typical misclassified images in out-sample testing: (a) Cluster predicted as apoptotic cell; (b) unattached cell predicted as attached cell; (c) unattached cell predicted as attached cell; (d) attached cell predicted as cluster; (e) dynamic blebbing cell predicted as attached cell; (f) dynamic blebbing cell predicted as cluster; (g) debris predicted as apoptotic cell; (h) debris predicted as dynamic blebbing cell. (Note: Recovered images are obtained from the autoencoder network)

98.30% recall. However, it performed worst for unattached cells. Unattached cells are generally easy to identify as shown in Figure 23c. From the typical misclassified images in out-sample testing as shown in Figure 32, we concluded that the blurring effects in the autoencoder network might be the cause for misclassifications. As shown in Figure 32(b) and 32(c), two unattached cells were blurred out after passing through the autoencoder network. Therefore, those cells looked similar to the attached cells visually. Moreover, this blurring effect might be more significant on the hidden representation generated by the decoder which was used to build the subnetworks.

### 5.4 Conclusion

Automated classification of hESC in phase contrast videos is essential for a fast quantifiable analysis of hESC behaviors. The proposed random network (RandNet) utilized unlabeled data for pre-training, and it incorporated both transfer and ensemble learning concepts. RandNet not only has lower training cost with pre-trained models, but

it also can improve performance through fine-tuning with labeled data. It had low performance variance in the cross validation results. This chapter demonstrated that RandNet is an efficient and effective method. In term of efficiency, it uses the combination of subsampling and pre-trained models to generate subnetworks. In term of effectiveness, it is a robust method that provides a generalized solution for hESC classification. Our objective in this paper has been to show that we can use both labeled and unlabeled datasets. This software enables quantitative analysis of changes/behavior of hESC in video. In the future, we will explore additional deep networks for building subnetworks. Since the blurring effects of the current simple network had affected classification performance, we will explore deeper network to learn a finer hidden representation for hESC classification.

# CHAPTER 6 Conclusion and Future Work

The advent of machine learning makes automated, fast and quantifiable analysis possible. Human embryonic stem cell research has been pushed to the next level with software such as CL-Quant. Techniques in segmentation, detection and classification are becoming more important for a quantifiable and reliable analysis. Conventional methods such as, classification and regression trees (CART), random forest (RF), neural network (NN) and support vector machine (SVM) have enabled statistically significant analysis in many researches. Now, the state-of-art deep learning with different convolutional neural network (CNN) architectures is becoming a main stream for further the analysis. It will help us dive deeper in understanding of human embryonic stem cell characteristics. In this paper, we proposed detection technique for cell regions, segmentation method for bleb analysis and deep learning architecture for classification as individual chapters.

Cell region detection is essential when analyzing the growth of cell cultures under different assays. For instance, Chapter 3 talks about quantifying the rate of hESC colony growth for different brand of cigarettes. Cell region detection can accelerate the study for evaluating the toxicity of cigarette smoke from conventional and harm reduction cigarettes. Fast analysis in toxicity test can help prevent harm and save lives. One of the cell region detection challenges observed in Chapter 4 is separating the close-knitted regions. As the result, future study on cell region detection will address this issue with deep learning techniques.

For bleb analysis as discussed in Chapter 4, we found out that image quality under 20x objective might not be sufficient for detail analysis of the bleb. Under 20x objective,

we can analysis the bleb area and intensity changes over time. However, the particles motion was not capture during bleb formation. It would be interesting to study the motion inside the bleb once the bleb lineage is captured in the form of segmented images. It would also us to quantify the particle motions in the bleb under different environment. As the result, I recommend bleb analysis with 40x objectives. For Future work, we should explore fast bleb detection and bleb sequence segmentation in videos with 40x objective.

Human embryonic stem cell classification is important to analyze the four intrinsic cell types as discussed in Chapter 5. It is important to biologists to understand the different type of cells in the assay at a given time. Therefore, a classification system can assist biologists in quantifying the cell types in each assay under different chemical environment. As observed in Chapter 5, unattached cells can transform into attached cells and attached cell can undergo either dynamic or apoptotic blebbing. Such transformation transitions can sometimes confuse biologists when categorizing them. Therefore, it is important to develop a probabilistic prediction model for each cell types. For future study, we will look into multi-labels for each cell with deep learnings.

# REFERENCES


[1] T. D. Miller and P. Elgard, "Defining modules, modularity and modularisation," Design for Integration in Manufacturing, Proceedings of the 13th IPS Research Seminar, Fugsloe, Aalborg University, 1998.

[2] G. Klushin, C. Fortin and Z. Tekic, "Modular design guideline for projects from scratch," in *29th DAAAM International Symposium*, Vienna, Austria, 2018.

[3] L. Correa, F. Kubota and P. Miguel, "Towards a contribution to modularity concepts and principal domains," *Product Management & Development,* vol. 10, pp. 119-130, 2013.

[4] S. Tatiraju and A. Mehta, "Image segmentation using k-means clustering, EM and normalized cuts," UC Irvine, 2008. [Online]. Available: http://www.ics.uci.edu/~dramanan/teaching/ics273a_winter08/projects/avim_report.pdf.

[5] K. Alsabti, S. Ranka and V. Singh, "A efficient k-means clustering algorithm," *Proc. 1st Workshop on High Performance Data Mining,* 1998.

[6] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman and A. Y. Wu, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Transactions on PAMI,* pp. 881-892, 2002.

[7] R. Farnoosh and B. Zarpak, "Image segmentation using Gaussian mixture model," *International Journal of Engineering Science,* vol. 19, pp. 29-32, 2008.

[8] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for Gaussian mixture," *Neural Comp.,* pp. 129-151, 1996.

[9] S. Gopinath, Q. Wen, N. Thakoor, K. Luby-Phelps and J. X. Gao, "A statistical approach for intensity loss compensation of confocal microscopy images," *Journal of Microscopy,* vol. 230, pp. 143-159, 2008.

[10] B. X. Guan, B. Bhanu, P. Talbot and S. Lin, "Automated human embryonic stem cell detection," *Proc. 2nd IEEE International Conf. On Health Informatics, Imaging and System Biology,* pp. 75-82, September 2012.

[11] G. T. Charras, M. Coughlin, T. J. Mitchison and L. Mahadevan, "Life and times of a cellular bleb," *Biophysical Journal,* vol. 94, no. 5, pp. 1836-1853, March 2008.

[12] J. Tinevez, U. Schule, G. Salbreux, J. Roensch, J. Joanny and E. Paluch, "Role of cortical tension in bleb growth," *PNAS,* vol. 106, no. 44, pp. 18581-18586, November 2009.

[13] W. Strychalski and R. D. Guy, "A computational model of bleb formation," *Mathematical Medicine and Biology,* vol. 30, no. 2, pp. 115-130, February 2013.

[14] F. Y. Lim, Y. L. Koon and K. Chiam, "A computational model of amoeboid cell migration," *Computer Methods in Biomechanics and Biomedical Engineering,* vol. 16, no. 10, pp. 1-12, January 2013.

[15] B. X. Guan, B. Bhanu, P. Talbot, S. Lin and N. Weng, "Comparison of texture features for human embryonic stem cells with bio-inspired multi-class support vector machine," *IEEE International Conference on Image Processing,* pp. 4102-4106, October 2014.

[16] H. Niioka, S. Asatani, A. Yoshimura, H. Ohigashi, S. Tagawa and J. Miyake, "Classification of C2C12 cells at differentiation by convolutional neural network of deep learning using phase contrast images," *Human Cell,* vol. 31, no. 1, pp. 87-93, 2018.

[17] W. Xie, J. A. Noble and A. Zisserman, "Microscopy cell counting and detection with fully convolutional regression networks," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization,* vol. 6, no. 3, pp. 283-292, 2018.

[18] Y. H. Chang, K. Abe, H. Yokota, K. Sudo, Y. Nakamura, C. Y. Li and M. D. Tsai, "Human induced pluripotent stem cell region recognition in microscopy images using convolutional neural networks," *IEEE International Conference on Engineering in Medicine and Biology Society,* pp. 4058-4061, 2017.

[19] R. Theagarajan, B. X. Guan and B. Bhanu, "DeephESC: An automated system for generating and classification of human embryonic stem cells," *IEEE International Conference on Pattern Recognition,* 2018.

[20] R. Theagarajan and B. Bhanu, "DeephESC 2.0: Deep generative multi adversarial networks for improving the classification of hESC," *Plos One,* 2019.

[21] J. A. Thomson, J. Itskovitz-Eldor, S. S. Shapiro, M. A. Waknitz, J. J. Swiergiel, V. S. Marshall and J. M. Jones, "Embryonic stem cell lines derived from human blastocysts," *Science,* vol. 282, no. 5395, pp. 1145-1147, November 1998.

[22] J. Nichols and A. Smith, "The original and identity of embryonic stem cells,"

*Development,* vol. 138, pp. 3-8, January 2011.

[23] M. Stojkovic, M. Lako, T. Strachan and A. Murdoch, "Derivation, growth and applications of human embryonic stem cells," *Reproduction,* vol. 128, pp. 259-267, September 2004.

[24] P. Talbot and S. Lin, "Mouse and human embryonic stem cells: can they improve health by preventing disease?," *Current Topics in Medicinal Chemistry,* vol. 11, no. 13, pp. 1638-1652, 2011.

[25] S. Lin, S. Fonteno, J. Weng and P. Talbot, "Comparison of the toxicity of smoke from conventional and harm reduction cigarettes using human embryonic stem cells," *Toxicological Sciences,* vol. 118, pp. 202-212, November 2010.

[26] S. Lin, S. Fonteno, S. Satish, B. Bhanu and P. Talbot, "Video bioinformatics analysis of human embryonic stem cell colony growth," *JOVE,* vol. 39, May 2010.

[27] "CL-Quant," Nikon, [Online]. Available: http://www.nikoninstruments.com/News/US-News/Nikon-Instruments-Introduces-CL-Quant-Automated-Image-Analysis-Software. [Accessed July 2013].

[28] S. Lin and P. Talbot, "Methods for culturing mouse and human embryonic stem cells," *Methods Mol Biol.,* vol. 690, pp. 31-56, 2011.

[29] J. Bushberg, J. Seibert, E. Leidholdt and J. Boone, The Essential Physics of Medical Imaging: Second Edition, Philadelphia, PA: Lippincott William & Wilkins, 2002.

[30] R. C. Gonzalez and R. E. Woods, Digital Image Processing: Third Edition, Upper Saddle River, NJ: Pearson Education Inc., 2008, pp. 689-794.

[31] M. Pepe, G. M. Longton and H. Janes, "Comparison of receiver operating characteristics curves," UW Biostatistics Working Paper Series- Working Paper 323 [eLetter], January 2008. [Online]. Available: http://biostats.bepress.com/uwbiostat/paper323.

[32] "Biostation-IM," Nikon, [Online]. Available: http://www.nikoninstruments.com/Vyrobky/ Cell-Incubator- Observation/BioStation-IM. [Accessed 30 November 2012].

[33] S. Technologies, "mTeSR Medium," January 2014. [Online]. Available: http://www.stemcell.com/en/Products/Popular-Product-Lines/mTeSR-TeSR2.aspx.

[34] R. V. Hogg and J. Ledolter, Engineering Statistics, New York: MacMillan, 1987.

[35] Z. Zhu and D. Huangfu, "Human pluripotent stem cells: an emerging model in developmental biology," *Development,* vol. 140, no. 4, pp. 705-717, February 2013.

[36] B. X. Guan, B. Bhanu, P. Talbot and S. Lin, "Detection of non-dynamic blebbing single unattached human embryonic stem cells," *IEEE International Conference on Image Processing,* pp. 2293-2296, September 2012.

[37] R. Adams and L. Bischoff, "Seeded region growing," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 16, no. 6, pp. 641-647, June 1994.

[38] Y. Sun and B. Bhanu, "Reflection symmetry-integrated image segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 34, no. 9, pp. 1827-1841, September 2012.

[39] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 888-905, August 2000.

[40] D. Comaniciu and P. Meer, "Mean Shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 5, pp. 603-619, May 2002.

[41] P. Meer and B. Georgescu, "Edge detection with embedded confidence," *IEEE Trans. Pattern Anal. Machine Intelligence,* vol. 23, no. 12, pp. 1351-1365, December 2001.

[42] C. Christoudias, B. Georgescu and P. Meer, "Synergism in low level vision," *IEEE 16th International Conference on Pattern Recognition,* vol. 4, pp. 150-155, 2002.

[43] F. Meyer, "Topographic distance and watershed lines," *Signal Processing,* vol. 38, pp. 113-125, July 1994.

[44] P. Soille, Morphological Image Analysis: Principles and Applications, Springer-Verlag, 1999, pp. 170-171.

[45] E. W. Weisstein, "Least Squares Fitting," From MathWorld--A Wolfram Web Resource, [Online]. Available: http://mathworld.wolfram.com/LeastSquaresFitting.html.

[46] P. Talbot, N. zur Nieden, S. Lin, I. Martinez, B. X. Guan and B. Bhanu, "Use of video bioinformatic tools in stem cell toxicology," *Handbook on Nanotoxicology,*

*Nanomedicine and Stem Cell Use in Toxicology,* April 2013.

[47] D. W. Shattuck, S. R. Sandor-Leahy, K. A. Schaper, D. A. Rottenberg and R. M. Leahy, "Magnetic resonance image tissue classification using a partial volume model," *NeuroImage,* vol. 13, pp. 856-876, 2001.

[48] B. X. Guan, B. Bhanu, P. Talbot and S. Lin, "Bio-driven cell region detection in human embryonic stem cell assay," *ACM/IEEE Trans. on Computational Biology & Bioinformatics,* vol. 11, no. 3, pp. 604-611, 2014.

[49] B. X. Guan, B. Bhanu, N. Thakoor, P. Talbot and S. Lin, "Human embryonic stem cell detection by spatial information and mixture of Gaussians," *IEEE HISB,* pp. 307-314, 2011.

[50] L. Breiman, J. Friedman, R. Olshen and C. Stone, Classification and Regression Trees, CRC Press, 1984.

[51] L. Breiman, "Random Forests," *Machine Learning,* vol. 45, no. 1, pp. 5-32, 2001.

[52] P. Geurts, D. Ernst and L. Wehenkel, "Extremely randomized trees," *Machine Learning,* vol. 63, no. 1, pp. 3-42, 2006.

[53] J. Morgan, "Classification and regression tree analysis," Boston University, Boston, MA, 2014.

[54] W. Wang, Y. Huang, Y. Wang and L. Wang, "Generalized autoencoder: a neural network framework for dimensionality reduction," *CVPR Workshops,* pp. 496-503, 2014.

[55] J. E. Sklan, A. J. Plassard, D. Fabbri and B. A. Landman, "Toward content based image retrieval with deep convolutional neural networks," *SPIE Medical Imaging International Society for Optics and Photonics,* p. 94172C, 2015.

[56] Z. Camlica, H. R. Tizhoosh and F. Khalvati, "Autoencoding the retrieval relevance of medical images," *Image Processing Theory Tools and Applications (IPTA) the fifth International Conference,* pp. 1-6, 2015.

[57] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *In Advances in Neural Information Processing Systems,* pp. 1097-1105, 2012.

[58] O. Pons, "Bootstrap of means under stratified sampling," *Electronic Journal of Statistics,* vol. 1, pp. 381-391, 2007.

[59] L. Rokach, "Ensemble-based classifiers," *Artificial Intelligence Review,* vol. 33, pp. 1-39, 2010.

[60] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012. [Online]. Available: https://arxiv.org/abs/1212.5701.

[61] S. J. Reddi, S. Kale and S. Kumar, "On the convergence of adam and beyond," *International Conference on Learning Representations,* pp. 1-23, 2018.

[62] D. M. W. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness, and correlation," *Journal of Machine Learning Technologies,* vol. 2, no. 1, pp. 37-63, 2011.

[63] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," *IEEE CVPR,* pp. 770-778, 2016.

[64] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR,* pp. 1-14, 2015.